MATH1222 - Introduction aux processus stochastiques

Projet Chaînes de Markov en temps discret

Groupe s180489-s170220: Martin RANDAXHE, Cyril RUSSE

Table des matières

1 Chaînes de Markov

1.1 Représentation de graphes

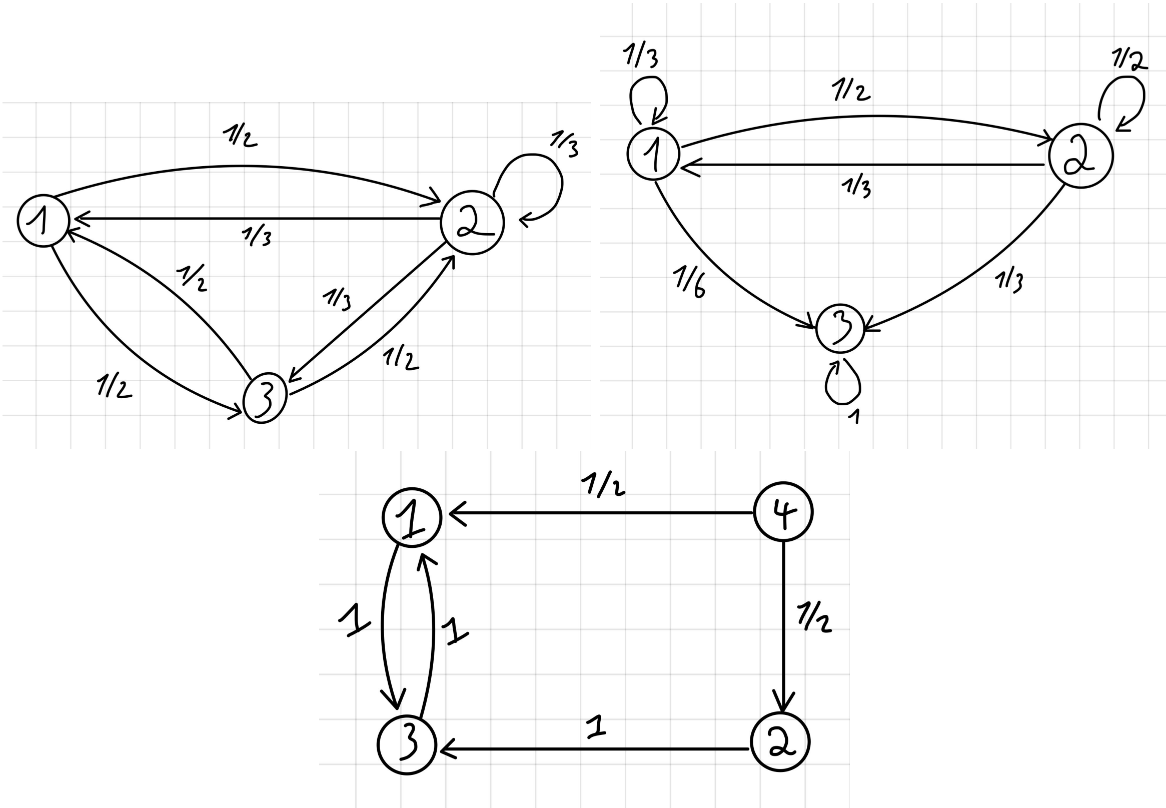

FIGURE 1 – Graphes de transition associés aux matrices de transition de l'énoncé de la question1

1.2 Simulation de réalisation des chaines de Markov

Nous avons créé différentes simulations pour chaque matrice en partant d'états initiaux différents. (tous disponibles dans le dossier "Graphique" avec les données correspondantes dans le dossier "Données") L'issue des simulations pour chaque simulation reste fortement similaire en fonction de l'état initiale donc nous en présenterons ici que une seule par chaine de Markov.

1.2.1 Observations

1. **Première matrice**
   Peu importe l'état initiale pour cette chaine car elle est régulière donc la chaine passera toujours par tous les états possibles.
2. **Deuxième matrice**
   Celle-ci possède 1 classe de passage comprenant l'état 1 et 2, et une classe finale comprenant l'état 3. Dorénavant, les simulations sont assez similaires si l'on prend l'état 1 ou 2 comme point de départ. Seule différence si l'on démarre de l'état 3 est que celui-ci est absorbant donc il ne passera en aucun cas pas les 2 autres états. Mais les 2 autres possibilités d'états initiaux, termineront toujours dans l'état 3 également.

3. **Troisième matrice**
Cette dernière possède, comme pour la seconde, une classe finale mais cette fois-ci composée de deux états : 1 et 3. Il n'est possible de passer qu'une seule fois par les états 2 et 4, si l'on venait à commencer dans l'un d'eux étant donné qu'il n'existe pas de probabilité de revenir dans le même état au pas de temps suivant. Nous obtenons alors comme résultat final une oscillation entre l'état 1 et 3 à chaque pas de temps.

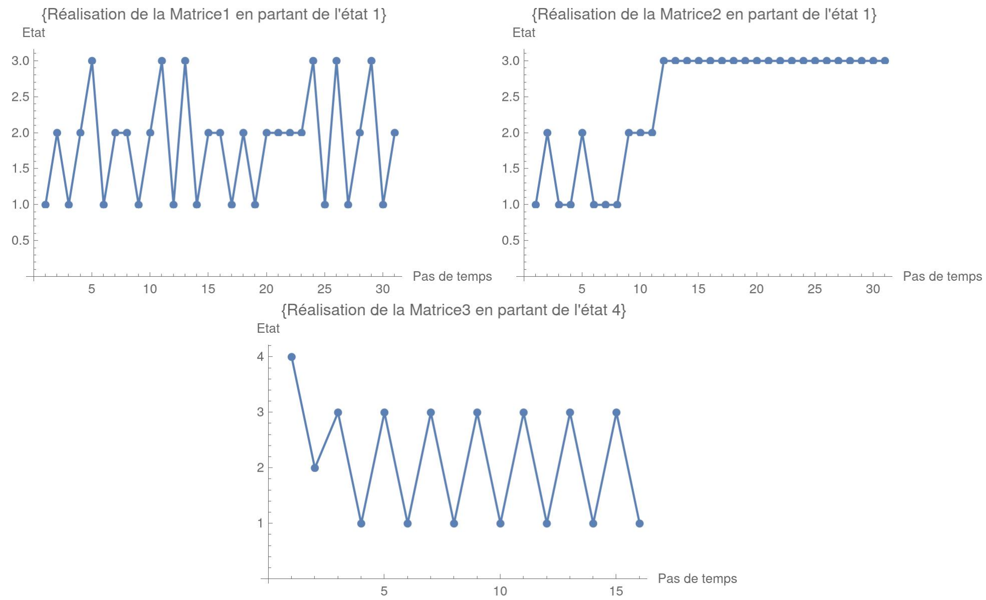

FIGURE 2 – Simulations de réalisation des chaînes de Markov

1.3 Calcul de l'évolution d'une distribution pour chaque état en fonction du temps

Nous avons donc calculé l'évolution en opérant le produit matriciel entre notre distribution initiale et la matrice de transition de la chaine étudiée. Et ainsi de suite à chaque fois avec le vecteur nouvellement créé à l'aide de ce produit matriciel. Soit, faire à la chaine, w = wP avec initialement w qui contient notre distribution initiale.

1.3.1 Première chaîne de Markov

Pour celle-ci, la distribution n'a aucune importance. Dans tous les cas, on observe que l'on obtient environ 30% dans l'état 1 et 3, et le reste, soit environ 40%, dans l'état 2.

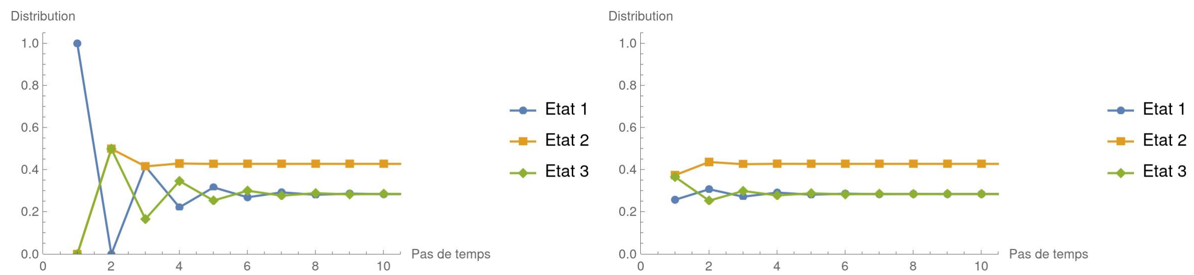

FIGURE 3 – Evolution du vecteur pour la matrice 1 avec comme distribution initiale [1,0,0] à gauche et aléatoire à droite

1.3.2 Deuxième chaîne de Markov

Une nouvelle fois, la distribution initiale n'a pas d'importance sur la distribution vers laquelle tend la chaîne de Markov. L'entiereté de la distribution finit toujours par se retrouver dans l'état 3. Au niveau de l'évolution que l'on peut observer graphiquement, les plus grosses modifications sont liées à la proportion initiale déjà présente dans l'état 3 qui y restera toujours.

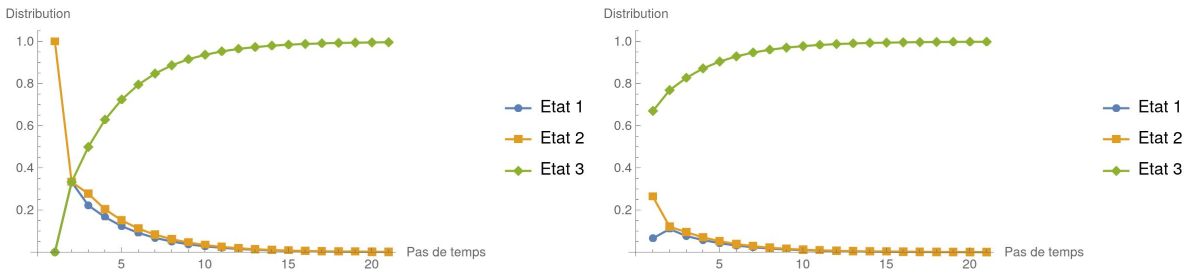

FIGURE 4 – Evolution du vecteur pour la matrice 2 avec comme distribution initiale [0,1,0] à gauche et aléatoire à droite

1.3.3 Troisième chaîne de Markov

Cette chaine-ci va toujours voir l'entiereté de sa distribution se retrouver, après au plus deux pas de temps répartie entre l'état 1 et 3. Ces proportions vont ensuite alterner entre ces deux états mais rester les mêmes. La distribution a dès lors une importance primordiale sur la répartition de la distribution entre ces deux états. Nous pourrions d'ailleurs déterminer dans quel état un individu se trouvera à un pas de temps t, juste en sachant son état initiale et en sachant en fonction de ça si t est pair ou impair.

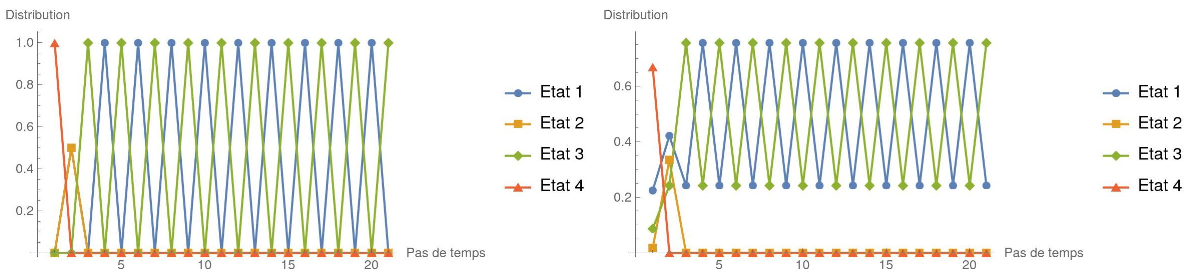

FIGURE 5 – Evolution du vecteur pour la matrice 2 avec comme distribution initiale [0,0,0,1] à gauche et aléatoire à droite

1.4 Evolution de la matrice P^t

On observe dans les trois cas une évolution très rapide vers une matrice ayant toutes leur lignes similaires. Chacune de ses lignes correspondent à la distribution invariante. Encore une fois, la troisième chaine de Markov étant un peu différente, après la puissance 3, cette distribution oscille entre l'entiereté dans 1 ou dans 3. En effet, dans mathematica si l'on essaye de le prouver en analysant la valeur de puissance tendant vers l'infini, on obtient des données indéterminées entre l'état 1 et 3. Il faut un peu plus de pas de temps pour la Matrice 2 pour tendre vers quelque chose de concordant avec cette distribution invariante, mais qui au bout d'un peu moins de la vingtième puissance tend très clairement vers [0,0,1]. Nous avons calculé différentes puissances dans notre fichier mathematica où l'on observe clairement ces évolutions.

$$
\begin{pmatrix} \frac{2}{7} & \frac{3}{7} & \frac{2}{7} \\ \frac{2}{7} & \frac{3}{7} & \frac{2}{7} \\ \frac{2}{7} & \frac{3}{7} & \frac{2}{7} \end{pmatrix}
\begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix}
\begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}
\begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}
$$

FIGURE 6 – (1)matrice 1 (2)matrice 2 (3)matrice3^3 (4)matrice3^4

1.5 Calcul des distributions invariantes

Cela revient à calculer w tel quel wP = w avec P, la matrice de transition. Et avec la somme des éléments du vecteur w égale à 1.

Soit par exemple pour la matrice 1 : $\begin{cases} y*1/3+z*1/2=x \\ x*1/2+y*1/3+z*1/2=y \\ x*1/2+y*1/3=z \\ x+y+z=1 \end{cases}$

Résolutions que nous avons ensuite réalisées à l'aide de Mathématica :

EXO 5

```
Solve[(Dot[{x, y, z}, p1] == {x, y, z}) && (x + y + z == 1)]
résous      produit scalaire
Solve[(Dot[{x, y, z}, p2] == {x, y, z}) && (x + y + z == 1)]
résous      produit scalaire
Solve[(Dot[{x, y, z, u}, p3] == {x, y, z, u}) && (x + y + z + u == 1)]
résous      produit scalaire
```

$\{\{x \to \frac{2}{7}, y \to \frac{3}{7}, z \to \frac{2}{7}\}\}$

$\{\{x \to 0, y \to 0, z \to 1\}\}$

$\{\{u \to 0, x \to \frac{1}{2}, y \to 0, z \to \frac{1}{2}\}\}$

FIGURE 7 – Distributions invariantes

Avec p1, p2, p3 contenant respectivement les matrices 1, 2 et 3.

1.5.1 Interprétations

Les valeurs de distributions invariantes obtenues concordent parfaitement avec les résultats obtenus pour les questions 3 et 4. Déjà dans ces questions-là, nous avions évoquer le fait que celà semblait correspondre à des distributions invariantes. Observations qui se trouvent encore une fois vérifiées. La 3ème chaîne de Markov se retrouve juste avec 1/2 pour les états 1 et 3. En effet, dans les précédentes questions nous avions précisé que l'entiereté de la distribution se répartissait quoi qu'il arrive dans ces 2 états. La distribution invariante nous prouve bien qu'à l'infini nous arions autant de chance de se trouver dans 1 que dans 3. Seulement dans nos autres observations, il apparaisait clairement qu'au vue de la structure de cette classe finale, en réalité la distribution ne va pas forcément tendre vers 50% dans chacun de ces états. Par contre, il est claire que le temps total passé dans chacun de ces 2 états sera égal. Il faut donc bien faire la distinction du fait que une autre chaine de Markov ayant 2 classes finales aurait pu avoir une même distribution invariante mais celà aurait été dû à la probabilité d'atteindre respectivement ces classes, qui seraient égales.

1.6 Proportion de temps passer dans chaque état

Afin de calculer les proportions de temps passé dans chaque état, nous avons créé un programme(Exo6.py) qui va, pour chaque état, créer jusqu'à T des simulations pour chaque t allant de 0 à T pour lesquels nous gardons en mémoire le nombre de passage par cet état divisé par t. Afin d'obtenir la propotion de temps passé dans cet état. Encore un fois, les résultats tendent vers les distributions invariantes.

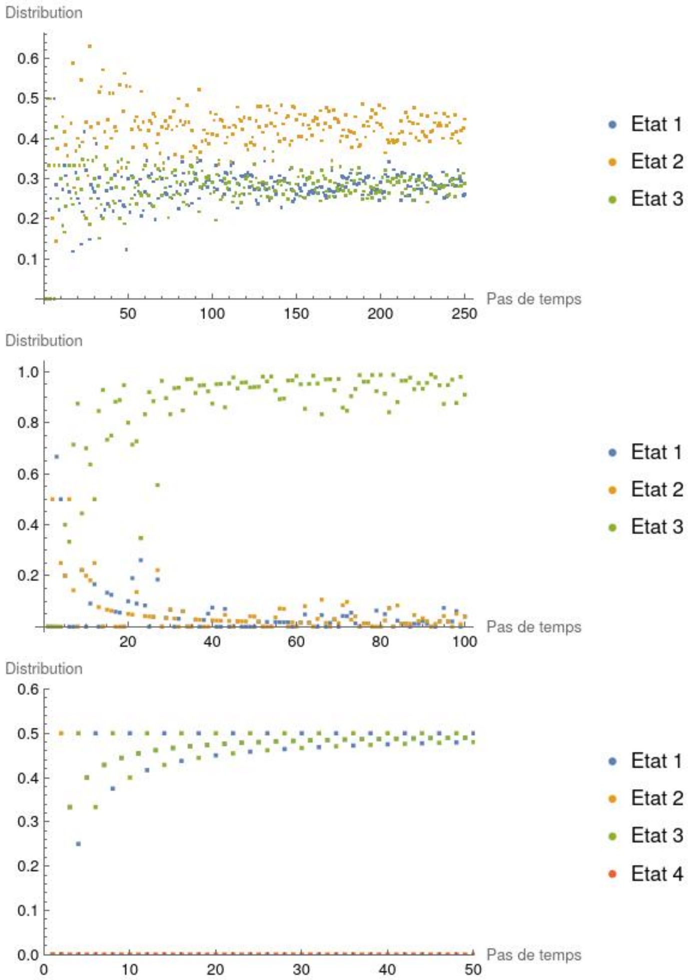

FIGURE 8 – Simulation des temps passer dans chaque état pour respectivement les matrices 1, 2 et 3

1.7 Temps moyen d'atteinte

Le programme Exo7.py génère un grand nombre de simulation de la deuxième chaîne de Markov partant de l'état 1 et qui s'arrête lorsqu'il atteint l'état 3 et donne la nombre de pas de temps pour y arriver. Enfin, on calcule la moyenne en divisant la somme de ces valeurs par le nombre de simulation ayant été executée. La valeur obtenue est 4,2.

2 Monopoly

2.1 Modélisation du problème par une chaîne de Markov

Le principal problème à résoudre est de définir les états de la chaîne de Markov. Il faut toutes les cases sur lesquels on a le potentiel de rester après avoir jouer un tour. Il est donc nécessaire d'éliminer les cases qui nous projettent instantanément dans une autre. Ensuite, nous allons définir deux états supplémentaires pour imaginairement représenter les tours où l'on reste en prison. Dans notre cas, les cases "chance" et "chancellerie" ne sont pas représentées par un état en tant que tel. C'est également le cas pour la case qui envoie directement le joueur en prison. Ca nous fait donc au total : trois cases "chance", trois cases "chancellerie" et la case de prison en moins. Soit 33 états mais à celà nous rajoutons 2 états qui représentent les tours en prison, donc 35 états.

Vous trouverez ci-dessous, une représentation des états par rapport au plateau et un exemple des flèches pour l'état 9 représentant le premier lancer de dé pour sortir de prison.

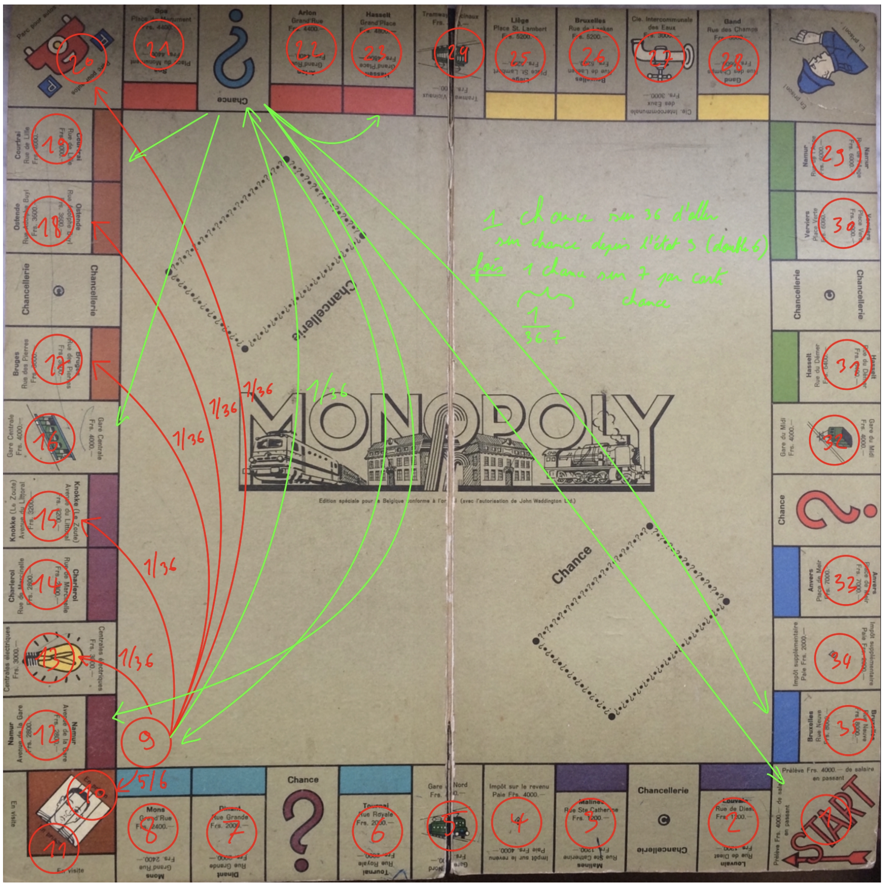

FIGURE 9 – Représentation de la chaîne de Markov sur le plateau

2.2 Calcul de la matrice de transition

Comme définie dans le précédent point, nous avons décidé d'utiliser 35 états. Ceux-ci représenteront dans l'ordre les cases, comme si le plateau ne contenait pas les cases "chance" et "chancellerie", ni celle qui envoie le joueur en prison. Les deux états supplémentaires pour la prison, seront les états juste avant la case de visite de prison. Elles représenteront respectivement les premiers et deuxièmes tours passés en prison. La visite libre représente également le troisième tour. Il est inutile de considérer l'ajout d'un état de plus étant donné que malgré le cout supplémentaire que le joueur devra déverser dans le cadre des règles, celà n'a aucune différence d'un point de vue du processus stochastique par rapport à un tour joué depuis la case de visite. Nous avons créé un programme en C permettant de créer la matrice sous forme d'un tableau en deux dimensions et nous la sauvegardons dans un fichier csv. Pour ensuite calculer les valeurs de cette matrice, nous avons commencé par créer un tableau répertoriant les différentes probabilité d'obtenir, lors d'un lancer de dés, les valeurs possibles, soient de deux à onze. A partir de là, pour chaque ligne de la matrice, nous savons quel état elle réprésente, et nous calculons donc pour chaque valeur de lancer de dé, à quelle case celà correspondrait. Ensuite, si cette nouvelle case n'a rien de particulier, la valeur de probabilité pour y aller correspondra à la valeur de probabilité d'obtenir cette valeur comme lancer de dés. Si par contre, la valeur de lancer de dés correspond à une case particulière, les différentes cases vers lesquelles nous renvoit celle-ci se verront obtenir une probabilité équivalente au lancer de dés multiplié par la probabilité d'être renvoyé sur la case en question. Sur le schéma représenté plus haut, nous avons un exemple dans le cas où l'on fait un lancer de dés égal à douze. Ce lancer de dés correspond à l'unique lancer 6 : 6, ayant une probabilité de 1/36. Nous allons donc répartir cette probabilité sur les sept autres cases vers lesquelles les cases "chance" sont susceptibles de nous envoyer. Soit pour chacune d'entre elles (1/36)*(1/7). Ces probabilités doivent être incrémentées à la valeur déjà présente de probabilité. En effet, il est possible de pouvoir tomber à la fois sur une case grâce à la valeur du lancer de dés, ou également à l'aide d'une carte chance.(Ce n'est pas le cas, dans notre exemple. L'état étant celui de la prison autorisant, uniquement d'aller d'en d'autres états que le 9, en cas de double)

2.3 Caractérisation de la chaîne de Markov

La chaine de Markov est composée d'une seule classe

— La chaine est périodique
— Etant donné l'unicité de la classe qui la compose, la chaine est irréductible
— Sa période étant supérieur à un, elle n'est pas apériodique et donc irrégulière

2.4 Calculs à partir de la matrice de transition

2.4.1 Evolution des probabilités $P(X_t = i)$

Nous avons réalisé dans mathematica le calcul des distributions w, en calculant à la chaîne w = wP avec P la matrice de transition. La distribution initiale correspond à 1 pour l'état 1 et 0 pour tout le reste. Nous avons sorti les graphiques en batonnet pour des valeurs de t égales à 5, 10, 15 et 30. On observe très peu de changement entre 15 et 30. Nous avons représenté ces données dans des graphiques en batonnet dans lesquels chaque batonnet représente un état.

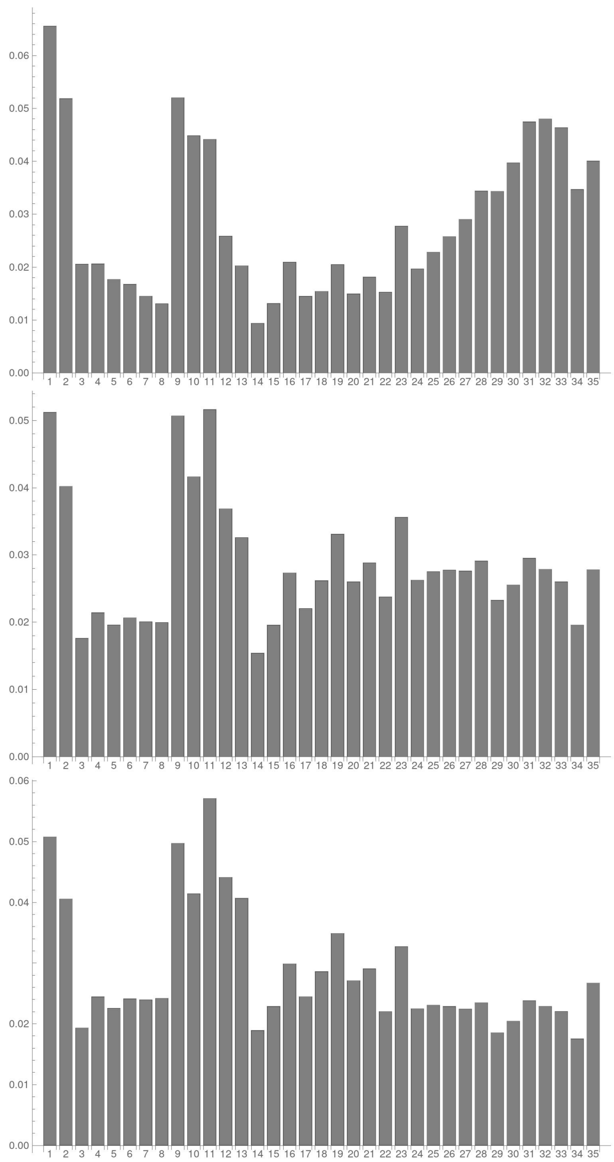

FIGURE 10 – Evolution de $P(X_t = i)$ pour $t = 5$(haut), $t = 10$(milieu) et $t = 30$(bas)

2.4.2 Proportion moyenne de temps passé par un joueur dans chacune des 40 cases

Celà revient à travailler avec la distribution invariante(vers laquelle on se rapproche avec les données des graphiques du point précédent). A l'unique différece qu'il faudrait rajouter les cases que l'on ne considère pas dans notre chaîne de Markov(chance, chancelerie...) mais pour lesquelles on retrouverait des valeurs nulles. Il faudrait également concaténer les valeurs de prison. Malheureusement nos précédents choix d'implémentations ne nous permettent pas de faire la distinction entre un troisième tour passé en prison et un joueur en visite.

2.4.3 Meilleure rue/gare

En observant nos graphiques du point 2.4.1, il en ressort que l'Avenue de la gare à Namur est la rue la plus rentable. On peut facilement l'expliquer par le fait qu'elle se trouve parmi les cases vers lesquelles une carte "chance" peut nous amener, et également n'est pas très loin du start par lequel on peut également se faire déplacer directement. De plus, elle n'est pas très loin d'une autre case(Louvain) à laquelle "chancellerie" peut nous amener. Les gares représentant dans nos graphiques les états 5, 16, 24 et 32. Il en ressort que la 16, soit la Gare centrale est la plus rentable des gares. Une nouvelle fois, au vue de son positionnement sur le plateau et du fait qu'une carte peut nous y amener, ce n'est pas surprenant.

2.4.4 Temps moyen d'atteinte

Pour les deux derniers points de cette question, nous avons repris et modifié un programme déjà utilisé pour le point 7 de la question une. Nous l'avons modifié et rendu plus pratique, il import la matrice depuis un fichier csv. Ensuite, le principe reste exactement le même que pour le point 7 de l'autre section.

— temps pour atteindre la prison depuis le start : 19,21

— temps moyen entre deux passages en prison : 20,09