TP : Chaîne de Markov

1 Étude de chaînes de Markov

— Charger avec la commande `load` : `markov.Rdata`.
La commande `ls()` permet de voir les données chargées et les fonctions.

— Pour simuler des chaînes de Markov, on utilise la fonction `markov(proba,nb,longueur,init)` où
  — `proba` est la matrice des probabilités de transition,
  — `nb` est le nombre de chaînes à simuler,
  — `longueur` est la longueur de chaîne simulée,
  — `init` est la condition initiale, si `init=0` les états initiaux sont choisis cycliquement, si `init=-1` les états initiaux sont choisis aléatoirement selon une loi uniforme, si `init` est un vecteur les états initiaux sont choisis en répétant le vecteur périodiquement.

— Avec la matrice `m3`, simuler 3 chaînes de longueur 20, en partant d'un état fixe.
— Simuler 30 chaînes en partant de l'état fixe puis d'une loi uniforme.
— Pour les différentes matrices, donner le nombre d'état, dessiner le graphe, étudier les simulations, quelles sont les caractéristiques de la chaîne.

— Pour visualiser les répartitions empiriques d'une chaîne, on utilise `(hm(m,direction,n)` où :
  — `m` matrice de chaînes de Markov simulées,
  — si `direction='t'`, la fonction affiche l'histogramme de la distribution de toutes les chaînes au temps `n`.
  — si `direction='c'`, la fonction affiche l'histogramme de la distribution de la chaîne numéro `n`.

— Pour les exemples à loi invariante uniforme, montrer avec cette fonction, la stationnarité lorsque l'on part de cette loi. Puis partir d'un état fixe pour montrer la stationnarisation ou la non stationnarisation selon les cas.

2 Ruine d'un joueur

Deux joueurs A et B jouent à pile ou face, chaque fois qu'un joueur gagne, il reçoit un euro de l'autre. Le jeu s'arrête lorsque l'un des deux est ruiné. Simuler un tel processus pour des valeurs de capital de départ fixé pour les 2 joueurs.

3 Modèle de Wright-Fisher

Ce modèle décrit l'évolution, au cours de générations successives et non chevauchantes, du nombre de copies d'un allèle a dans une population haploïde de taille finie N. À chaque temps t, on note n_t, ce nombre de copies. On suppose simplement que chacun des N individus/gènes d'une génération donnée tire son parent parmi les N individus de la génération précédente suivant une loi uniforme (tous les parents sont équiprobables) et que les tirages des N parents sont indépendants entre eux. Dans le cas où l'on modélise des possibles mutations de A vers a et de a vers A, chaque individu aura l'allèle de son parent avec probabilité $1 - \mu$ et l'autre allèle avec probabilité μ.

— Simulation du processus de Wright-Fisher avec mutation. Écrire un programme permettant d'obtenir une trajectoire du processus de Wright-Fisher sur une durée $tmax$ générations.

— Étude du nombre de copies de l'allèle a
  - — Simuler 100 trajectoires du processus pour $N = 10$ et $n_0 = 1$, $\mu = 0.01$, $tmax = 100$;
  - — Représenter les trajectoires et la loi empirique de n_t à chaque temps t
  - — En utilisant la matrice de transition de la chaîne de Markov, retrouver la loi exacte de n_t et comparez-la à la loi empirique.
  - — Quel est l'effet de n_0 ?

— Étude du processus de Wright-Fisher sans mutation :
  - — Simuler 50 trajectoires du processus pour $N = 10$ et $n_0 = 1$, $tmax = 100$ mais cette fois $\mu = 0$ et représentez la loi de de n_t à chaque temps t.
  - — Recommencez cette simulation.
  - — Quel est l'effet de n_0 ?
  - — Retrouvez ces résultats en utilisant la matrice de transition.

— Étude du processus de Wright-Fisher avec sélection (si on a le temps).
Modélisez le fait que l'allèle mutant a à une valeur sélective 1-s (A ayant une valeur sélective 1).

— Ergodicité.
Comparez la moyenne de 100 réalisations de n_{50} avec la moyenne des valeurs $\{n_{50}, n_{51} \ldots n_{149}\}$

sur une seule trajectoire puis avec la moyenne des valeurs $\{n_{50}, n_{150}, \ldots\}$. Étudiez les variances de ces moyennes.