Université Paris Saclay - UFR Sciences

Master 1 Bioinformatique et biostatistiques (BIBS)

2021-2022

Rapport de stage

Inférence démographique chez la levure *Saccharomyces cerevisiae*

Lindsay GOULET

Université Paris-Saclay
Laboratoire Interdisciplinaire des Sciences du Numérique (LISN)
UMR9015
Département Science des données
Équipe BioInfo

Encadrantes
Fanny POUYET - Flora JAY

02 mai 2022 - 24 juin 2022

Remerciements

Je remercie en premier lieu mes encadrantes de stage Flora Jay et Fanny Pouyet, pour leur confiance, le temps qu'elles ont décidé de m'accorder et les connaissances qu'elles ont su m'apporter.

Je remercie également les enseignants du Master M1 BIBS de Paris-Saclay pour les connaissances et notions qu'ils m'ont apportées. Je tiens à remercier spécialement Sarah Cohen-Boulakia pour m'avoir aidée dans les démarches pour trouver mon TER et ainsi mon stage.

Enfin, je remercie les membres de l'équipe BioInfo et du groupe PopGen, notamment Louis Ollivier avec qui j'ai travaillé en collaboration, pour toute l'aide qu'ils m'ont apportée.

Table des matières

1 Introduction

1.1 Présentation de l'équipe de recherche

Le travail présenté dans ce rapport à été réalisé au Laboratoire Interdisciplinaire des Sciences du Numérique (LISN). Ce laboratoire regroupe 16 équipes de recherche et est dirigé par Sophie Rosset, directrice de Recherche au CNRS.
Mon stage s'est déroulé dans le Département Sciences des Données et plus précisément dans l'équipe de Bioinformatique (BioInfo). Au cours de ce stage, j'ai également travaillé dans un groupe de recherche plus large, regroupant certains membres des équipes BioInfo et TAU (spécialisée dans le machine learning et l'optimisation) nommé PopGen et spécialisé dans la génétique des populations.
L'équipe Bioinfo est dirigée par Alain Denise et Sarah Cohen-Boulakia. Elle est composée de 8 enseignants-chercheurs, dont mes encadrantes de stage Flora Jay et Fanny Pouyet, ainsi que de non-permanents comme les doctorants, étudiants en stage ou étudiants en alternance, dont Louis Ollivier avec qui j'ai travaillé.

1.2 Contexte général

1.2.1 La génétique des populations

La génétique des populations est une branche fondatrice de la génétique moderne ; elle a été initiée par les biologistes Fisher, Haldane et Wright entre 1920 et 1940. Elle peut être définie comme l'étude de la distribution et des changements de la fréquence des versions d'un gène, appelées allèles, dans les populations d'êtres vivants, sous l'influence des pressions évolutives (sélection naturelle, dérive génétique, mutation, et migration). Elle se développe à partir des principes fondamentaux de la génétique mendélienne et de la théorie darwinienne de l'évolution. La génétique des populations a de nombreuses applications entre autre en épidémiologie, en agronomie ou en génétique de la conservation. En effet, la variabilité de la taille d'une population est un phénomène important à étudier et à considérer. Par exemple, il existe certaines espèces dont la taille de la population varie énormément : c'est le cas des espèces invasives comme les pucerons (*Aphidoidea*) ou encore le frelon asiatique (*Vespa velutina*) mais également des espèces en déclin (certaines populations de poissons par exemple). Il est donc important de considérer les variations de taille d'une population afin de savoir s'il existe une structuration de la population liée à des facteurs humains. La génétique permet également de comprendre l'influence, sur des populations, des variations environnementales (c'est-à-dire les variations qui touchent l'ensemble des éléments biotiques ou abiotiques qui entourent une espèce). C'est une notion fondamentale pour étudier les effets des phénomènes de domestication sur le génome d'une espèce.

1.2.2 *Saccharomyces cerevisiae*

Communément appelée "levure de bière" ou "levure de boulanger", *Saccharomyces cerevisiae* est une levure très largement utilisée dans l'industrie agro-alimentaire. Il s'agit d'un des organismes modèles eucaryotes étudiés en laboratoire [1].
Le génome de *Saccharomyces cerevisiae* a été le premier génome eucaryote entièrement séquencé (en 1996 [2]). Il a depuis été intensivement étudié et annoté, les informations le concernant étant gérées par la base de données Saccharomyces Genome Database [3].

Saccharomyces cerevisiae est donc très utilisée dans le monde entier, notamment pour le pain, la bière ou le vin par exemple depuis des dizaines de siècles. Les premières traces de fermenta-

tion utilisant la levure remontent à plus de 9000 ans et se situent en Chine[4]. La levure a donc été domestiquée par l'espèce humaine relativement tôt. En effet, la domestication de la levure se situe après celle du loup (il y a 15 000 ans), mais à la même période que le cochon (8 000 ans) ou le chat (9 500 ans) et bien avant le canard (1 000 ans)[5]. En biologie, la domestication d'une espèce correspond à l'acquisition et la transformation de caractères et de comportements héréditaires au contact de l'Homme, que ce soit suite à une interaction prolongée ou à un effort volontaire de sélection par l'Homme.

Le cycle de vie de la levure est dit haplodiplophasique, c'est-à-dire que la levure est capable de vivre sous forme diploïde (possédant deux versions de chaque chromosome) comme haploïde (une version). Elle vit préférentiellement sous forme diploïde car cela lui permet une meilleure résistance aux changements environnementaux et lui procure une meilleure résistance aux divers stress et se reproduit par bourgeonnement. Dans des conditions extrêmes, elle peut s'orienter vers un état de quiescence ou peut encore passer sous forme haploïde (cellules a et α) et sporuler. La fusion d'une cellule a et d'une cellule α lui permet par la suite de revenir à un état diploïde.

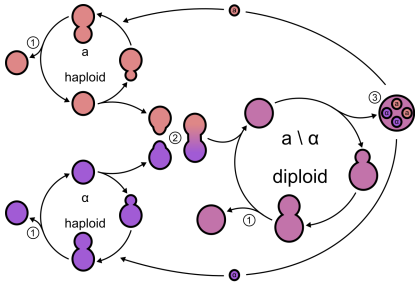

FIGURE 1 – **Cycle cellulaire simplifié de la levure**[6]

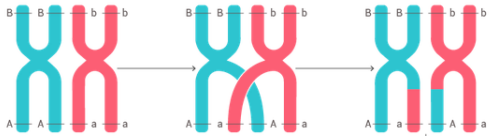

FIGURE 2 – **Schéma expliquant le fonctionnement du phénomène de recombinaison génétique**[7]

Au cours de la méiose, il est possible d'observer des phénomènes appelés recombinaison génétique. Il s'agit d'un échange d'information génétique entre deux génomes différents ou bien entre deux chromosomes. La recombinaison génétique, tout comme la mutation, est essentielle pour la diversité génétique puisque qu'elle permet de créer une infinité de combinaisons à partir d'un nombre de génomes restreint.
Il a récemment été montré que la levure ne se reproduirait par reproduction sexuée qu'environ toutes les 1 000 à 10 000 générations [8].
La reconstruction de l'histoire évolutive d'une espèce se basant sur la diversité observée actuellement, il est donc nécessaire de prendre en compte cette particularité dans son cycle de vie.

1.2.3 Objectifs

L'objectif de ce stage est donc de savoir si la fréquence de la recombinaison génétique a un impact sur l'inférence démographique de la levure et si les logiciels d'inférence démographique couramment utilisés sont capables de passer outre cette particularité. Pour répondre à cette question, nous allons commencer par simuler des populations panmictiques (tous les individus sont des partenaires potentiels avec la même probabilité) soumises à différentes fréquences de recombinaison (de toutes les générations à une fois toutes les 1000 générations) et soumises à différents évènements démographiques (expansion, goulots d'étranglement, etc.). Par la suite, nous allons inférer l'histoire évolutive de ces populations avec différents logiciels (notamment SMC++[9] et *dadi*[10]) et les comparer afin d'observer si la fréquence de recombinaison impacte l'histoire évolutive. L'objectif à long terme est de pouvoir utiliser le/les logiciel(s) d'inférence démographique sur les données réelles de la levure.

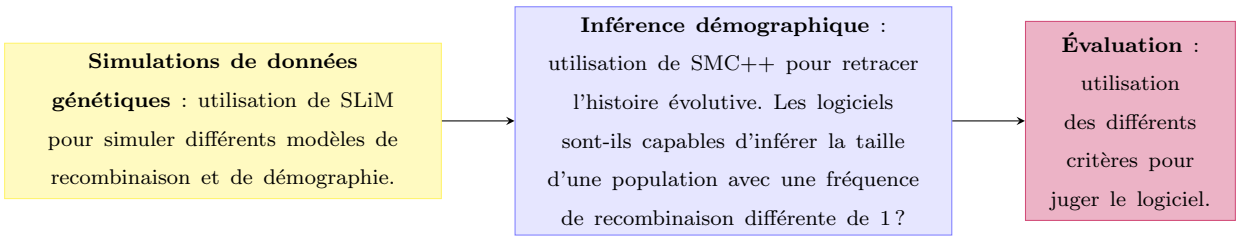

FIGURE 3 – **Pipeline global suivi durant le stage**

2 Matériels et méthodes

2.1 Simulations et modèles théoriques

2.1.1 Présentation de SLiM

SLiM[9] est un logiciel de simulation génétique dit "forward-in-time", c'est-à-dire qu'il reconstruit l'histoire du passé jusqu'aux générations récentes. SLiM est basé sur un langage appelé Eidos, spécialement conçu pour le logiciel (très proche du langage R). Par défaut, SLiM s'appuie sur un modèle d'évolution de Wright-Fisher, il admet en particulier certaines hypothèses ; en particulier, (1) les générations ne se chevauchent pas, (2) la probabilité qu'un individu ait une descendance est proportionnelle à sa *fitness* (valeur sélective), (3) les individus sont diploïdes, et (4) la descendance est générée par recombinaison des chromosomes parentaux avec ajout de nouvelles mutations.

Versions utilisées

SLiM	3.7.1
Eidos	2.7.1

2.1.2 Modèles théoriques

Afin de tester l'impact de la fréquence de recombinaison sur l'inférence démographique, nous allons utiliser trois modèles théoriques.

Le premier modèle étudié est un modèle très simpliste, c'est le modèle dit constant (Figure 4a). Il représente une population panmictique constante au cours du temps.
Le second modèle (Figure 4b) est le modèle à croissance exponentielle. La taille de la population est donc constante pendant $7N_{e,initial}$ générations puis subit une croissance exponentielle pendant $3N_{e,initial}$ générations.
Le troisième modèle (Figure 4c) est un modèle à deux époques avec une phase de réduction ("bottle neck" ou "goulot d'étranglement").

Ces modèles font apparaître la notion de taille efficace de la population (N_e), qui correspond à la taille de la population idéale panmictique nécessaire pour obtenir la même diversité

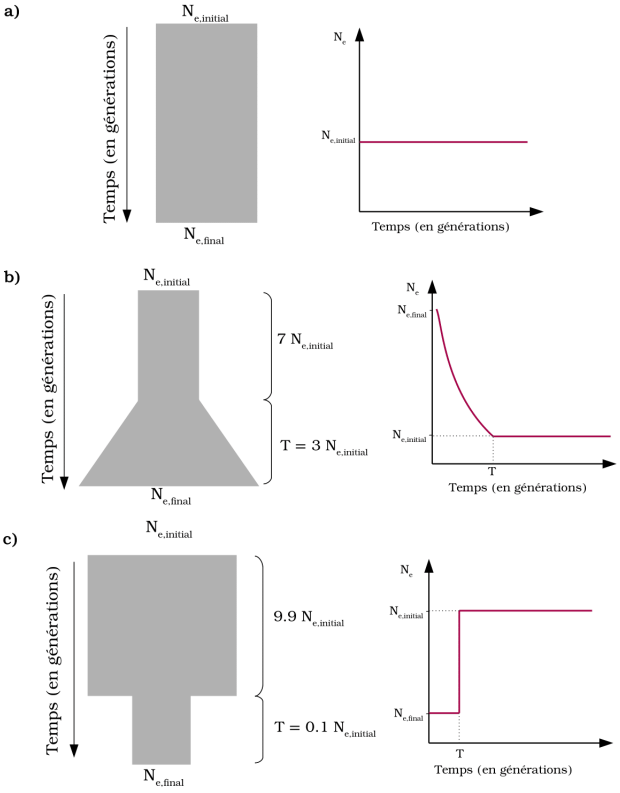

FIGURE 4 – **Schématisation des modèles théoriques et courbes attendues.**
a) Modèle constant. b) Modèle exponentiel. c) Modèle à 2 époques avec réduction.

génétique que celle observée. On introduit ici également le paramètre T qui correspond au temps écoulé depuis le phénomène étudié.

2.1.3 Paramètres utilisés

Afin de réaliser des simulations réalistes pour une population de levure tout en respectant un temps de calcul et un besoin en ressources raisonnables, il a été nécessaire de fixer plusieurs paramètres à l'aide de la littérature. Nous avons donc les paramètres suivants :

Modèle	Constant	Exponentiel	Réduction
Taux de mutation par génération μ	$1.67.10^{8}$		
Taux de recombinaison ρ	$0.5.10^{-8}$		
Taille du génome L	1 chromosome de 10 Mbp		
Taille de population efficace $N_{e,initial}$	2 000	1 000	2 000
Echantillon sélectionné n	20		
Nombre de générations G	10 000	10 000	20 000
Fréquence de recombinaison	varie entre 1 et 1/1000		
$\nu = \frac{N_{e,final}}{N_{e,initial}}$	1	15.7	0.1
T	$\times$	3000	200

Cette partie a été réalisée en amont de mon stage par Louis Ollivier.

2.1.4 Données utilisées

Les données génétiques utilisées au cours de ce stage proviennent donc exclusivement des simulations réalisées avec le logiciel SLiM. Les données obtenues via SLiM sont sous la forme de séquence
d'arbres (format *.trees*). Les séquences d'arbres permettent de suivre l'ascendance locale pour une portion précise d'un chromosome. On obtient donc, pour chaque intervalle le long d'un chromosome, un arbre retraçant la généalogie des individus à cette position. A chaque point de recombinaison cet arbre peut donc être différent. Il est également à noter que, pour des questions de performance, il est préférable de ne pas simuler les mutations neutres directement dans SLiM mais de les ajouter par la suite via le package msprime[12] dans Python. En effet, cela accélère les simulations car les mutations ne seront ajoutées que dans les régions du génome ayant été conservées.

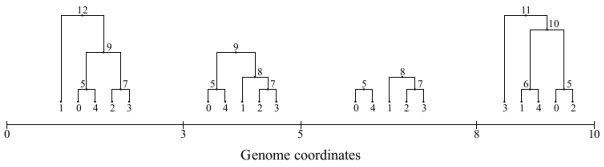

FIGURE 5 – **Schéma des différents arbres le long d'une séquence génomique**[11]

```
##fileformat=VCFv4.2
##fileDate=20220622
##source=SLiM
##INFO=<ID=MID,Number=.,Type=Integer,Description="Mutation ID in SLiM">
##INFO=<ID=S,Number=.,Type=Float,Description="Selection Coefficient">
##INFO=<ID=DOM,Number=.,Type=Float,Description="Dominance">
##INFO=<ID=PO,Number=.,Type=Integer,Description="Population of Origin">
##INFO=<ID=GO,Number=.,Type=Integer,Description="Generation of Origin">
##INFO=<ID=MT,Number=.,Type=Integer,Description="Mutation Type">
##INFO=<ID=AC,Number=.,Type=Integer,Description="Allele Count">
##INFO=<ID=DP,Number=1,Type=Integer,Description="Total Depth">
##INFO=<ID=MULTIALLELIC,Number=0,Type=Flag,Description="Multiallelic">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
#CHROM      POS   ID    REF   ALT   QUAL  FILTER      INFO
FORMAT     i0    i1    i2    i3
1          352   .     A     T     1000  PASS        MID=37010951;...
GT         0|0   1|0   0|0   0|0
1          653   .     A     T     1000  PASS        MID=31286779;...
GT         0|0   0|1   0|0   0|0
```

FIGURE 6 – Exemple de fichier .vcf

Par la suite, le package tskit[13] de Python permettra de convertir cette séquence d'arbres en fichier VCF (Variant Call Format). Les fichiers VCF sont couramment utilisés pour stocker les variations le long du génome (par exemple les SNPs, les indels, les variants structuraux, *etc.*) pour un certain nombre d'individus ainsi que des annotations les concernant.

Versions utilisées

Python	3.10.4	
numpy	1.22.4	Manipulation de matrices/tableaux et utilisation de fonctions mathématiques
pandas	1.4.2	Manipulation de fichiers .csv
pyslim	0.700	Manipulation des séquences d'arbres produits par SLiM
msprime	1.2.0	Simulation d'arbres généalogiques et de données de séquences génomiques (ajout des mutations neutres).
tskit	0.4.1	Outils pour l'utilisation/analyse de séquences d'arbres (notamment conversion en VCF).
matplotlib	3.1.2	Création de graphiques

2.2 Inférence démographique

2.2.1 Présentation de SMC++

SMC++[14], pour *Sequential Markov Coalescent + Plenty of Unlabeled Samples*, est un logiciel (en C++) permettant d'estimer l'histoire de la taille efficace des populations au cours du temps à partir de centaines de génomes entiers non phasés. SMC++ requiert un paramètre obligatoire (le taux de mutation par génération, μ) ainsi que des paramètres optionnels de régularisation qui, nous allons le montrer, affectent la qualité des ajustements obtenus.
Nous avons notamment observé l'effet du type de spline (cubic ou piecewise), la regularization penalty (1, 3, ou 6) ainsi que le nombre de nœuds de spline *knots* (10 ou 21).

Versions utilisées

SMC++	v1.15.4

2.2.2 Pairwise sequentially Markov coalescent

SMC++ repose sur le PSMC[15] (pairwise sequentially Markov coalescent). Cette méthode permet d'analyser les séquences génomiques non phasées d'un même individu diploïde (ou de deux individus haploïdes). Elle est basée sur les chaînes de Markov cachées. Un modèle de Markov caché est une paire de processus stochastiques, X_t et Y_t où X_t est le processus caché (ne peut pas être directement observé) et Y_t est le processus observé. À chaque instant t, X_t prend l'un des N états possibles (selon une distribution de probabilité donnée, appelée distribution de transition). Cet état est uniquement dépendant de l'état de X_{t-1} (processus de Markov). À chaque passage à un nouvel état, une valeur de Y_t est générée (selon une distribution de probabilité dépendante de X_t, appelée distribution d'émission). Dans le cas du PSMC, les états cachés correspondent aux généalogies locales à chaque locus. Les états possibles sont les temps de coalescence entre les 2 allèles étudiés. On appelle temps de coalescence le temps entre le présent et l'évènement de coalescence, c'est-à-dire la fusion entre deux lignées dans un arbre généalogique. Le temps est discrétisé en intervalles afin d'obtenir des états discrets et non continus.

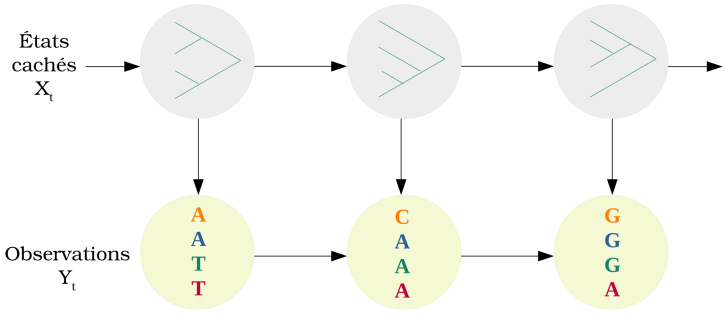

FIGURE 7 – **Schématisation d'une méthode de type SMC** (inspiré de [16])

2.3 Critères d'évaluation

2.3.1 Erreur entre les courbes (ISRE)

L'un des premiers critères d'évaluation du logiciel SMC++ que nous avons défini est l'ISRE (Integral of Squared Relative Error). Nous avons tout d'abord calculé une courbe moyenne sur tous les réplicats (pour un modèle précis et des paramètres SMC++ précis), considérant qu'un ensemble de réplicats correspond à un ensemble de régions indépendantes du génome contribuant tous à la prédiction. Nous avons ensuite calculé l'ISRE de la manière suivante :

$$ISRE = \frac{1}{G}\int_0^G \frac{\left(f_{att}(t) - f_{obs,moy}(t)\right)^2}{f_{att}(t)^2}dt$$

où G est le nombre de générations pour lesquelles l'histoire démographique a été reconstruite, f_{att} la fonction attendue du modèle étudié et $f_{obs,moy}$ la fonction moyenne observée obtenue avec SMC++.

2.3.2 Fit et estimation des paramètres démographiques

Le deuxième critère étudié concerne l'inférence des paramètres N_e, T et ν. Le logiciel SMC++ ne permet pas d'inférer directement ces paramètres. Nous avons donc défini une méthode permettant de les estimer à partir des graphes obtenus. Pour chacun des modèles étudiés et chaque réplicat, nous avons ajusté aux données des fonctions des formes attendues (Figure 4). À partir de ces fonctions, nous avons pu estimer chacun des paramètres. Les fonctions attendues pour chacun des modèles sont celles données en Figure 4. Cette procédure est détaillée en Annexe 1.

2.4 Pipeline détaillé

Le pipeline mis en place lors du stage a été le suivant :

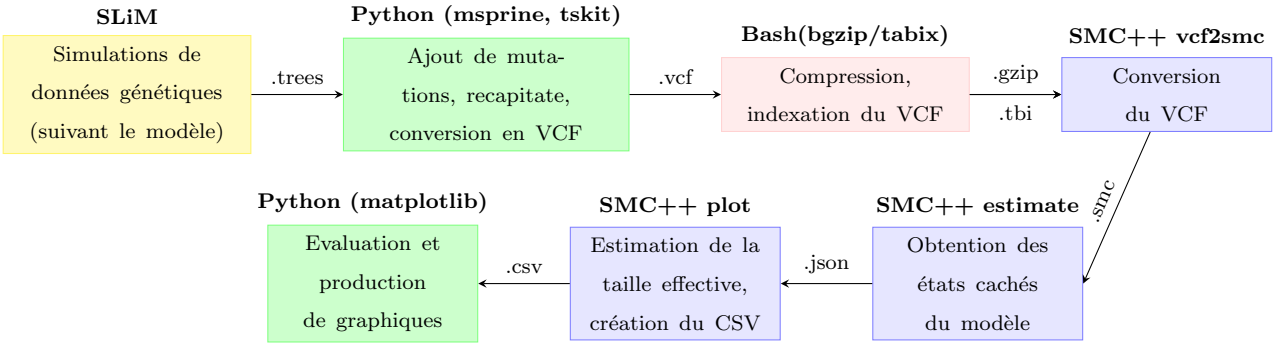

FIGURE 8 – **Pipeline détaillé suivi durant le stage**

La majeure partie du pipeline a été réalisée sur le cluster de calcul de l'IFB[17]. Seule la partie "évaluation et production de graphiques" a été réalisée sur ma machine personnelle.
Concernant la première partie du stage (le choix des hyper-paramètres de SMC++), nous avons réalisé 50 réplicats pour chacun des modèles et des paramètres testés.
Concernant la seconde partie (l'impact de la fréquence de recombinaison), nous avons réalisé 100 réplicats pour chacun des trois modèles avec les paramètres retenus.

Mes travaux sont accessibles via le lien suivant : https ://gitlab.inria.fr/ml_genetics/private/yeast-evolution

3 Résultats

3.1 Choix des hyper-paramètres pour SMC++

Avant de pouvoir utiliser SMC++ afin d'étudier l'impact de la fréquence de recombinaison, le premier objectif de ce stage a été d'identifier les hyper-paramètres (type de spline, régularisation et nombre de noeuds) optimaux pour le logiciel. Pour cela nous avons donc calculé l'ISRE, une mesure de la distance entre trajectoire réelle et prédite, pour chacun des trois modèles étudiés (Figure 4). On peut alors observer que pour le modèle constant, le jeu de paramètres ayant l'ISRE la plus faible est spline piecewise, rp 1, knots 10 ($ISRE = 2,4.10^{-}04$). Pour le modèle exponentiel, c'est spline cubic, rp 6 et knots 21 ($ISRE = 1,2.10^{-}01$). Pour le modèle à 2 époques c'est spline piecewise rp 1 knots 21 ($ISRE = 3,2.10^{-}01$). On observe ainsi qu'il n'y a pas de paramètres optimaux permettant de minimiser l'ISRE pour les 3 modèles étudiés. Certains jeux de paramètres produisent une ISRE faible pour un modèle étudié et une ISRE élevée pour d'autres modèles. Par exemple, le trio "cubic, rp 3, knots 21" a une ISRE faible pour le modèle exponentiel, mais très élevée pour le modèle constant.

La reconstruction d'une même histoire évolutive pour le modèle exponentiel avec les différents paramètres de SMC++ est détaillée en Annexe 2.

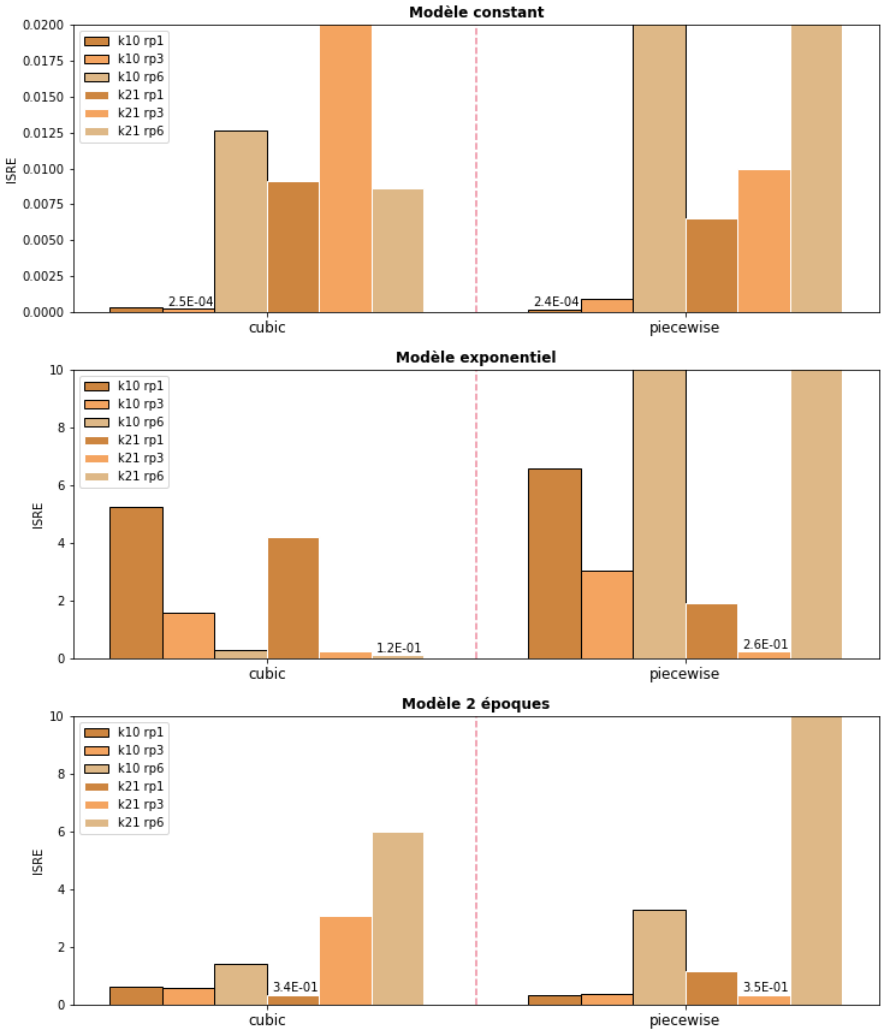

FIGURE 9 – **ISRE calculée pour chacun des modèles et pour différents paramètres.** k = knots (nombre de nœuds du modèle de spline), rp = regularization penalty

Au vu des résultats obtenus en Figure 9, nous avons sélectionné pour SMC++ les paramètres cubic, rp 3, knots 10. En effet, les résultats obtenus avec le paramètre *piecewise* formaient des courbes en escalier (donc discontinues) avec des intervalles temporels assez importants, ce qui nous a conduit à ne pas le sélectionner (la taille efficace N_e est continue au cours du temps et nous trouvions plus réaliste d'avoir une reconstruction continue). Nous avons donc sélectionné le paramètre *cubic*. Pour les paramètres *rp* et *knots*, nous avons sélectionné respectivement 3 et 10 puisque le jeu de paramètre "spline cubic, rp 3, knots 10" est celui ayant l'ISRE la plus faible si l'on moyenne sur les 3 modèles.

3.2 Analyse de l'impact de la fréquence de recombinaison

Le deuxième objectif du stage était donc d'étudier l'analyse de la fréquence de recombinaison sur l'inférence démographique (ici toutes les 1, 10, 100 ou 1000 générations). Étant donné que nous voulions observer l'impact de la fréquence, nous avons conservé un taux de recombinaison par génération identique dans tous les scénarios. Grâce à notre pipeline, nous avons réalisé l'inférence automatique de la démographie de 1200 jeux de données (3 modèles étudiés * 4 fréquences testées * 100 réplicats) avec les mêmes hyper-paramètres de SMC++ (spline cubic, rp 3, et knots 10). Nous observons globalement que, quelque soit le modèle démographique, l'erreur de reconstruction est élevée lorsque la recombinaison a lieu seulement toutes les 1000 générations (ISRE pour 1000GR = $0.009, 0.3$ et 0.65 pour le modèle constant, exponentiel et 2 époques respectivement ; Figures 10-11-12). Elle est en revanche du même ordre de grandeur pour les trois autres fréquences pour les modèles constant et exponentiel.

Concernant l'inférence des différents paramètres démographiques, ils sont globalement sous ou sur-estimés (sauf N_e pour le modèle constant, Figure 10). Pour l'estimation de N_e et de nu, on ne note pas de différence notable du biais entre les différentes fréquences de recombinaison pour le modèle exponentiel (Figure 11). En revanche, on observe clairement une différence du biais pour l'inférence de T pour le modèle exponentiel (Figure 11) et pour les trois paramètres du modèle à 2 époques (Figure 12).
On peut également noter qu'il n'y a pas de différence notable dans la variance des paramètres pour chacun des modèles.

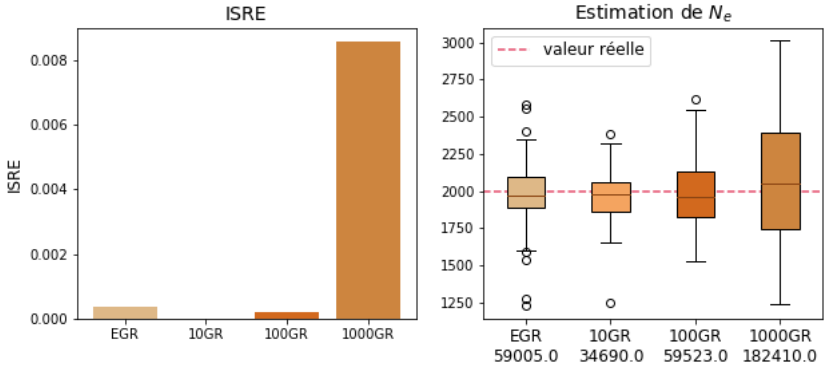

FIGURE 10 – **ISRE calculée et estimation des paramètres du modèle constant**
EGR : recombinaison toutes les générations, 10GR : recombinaison toutes les 10 générations, 100GR : recombinaison toutes les 100 générations, 1000GR : recombinaison toutes les 1000 générations. Les chiffres correspondent à la variance.

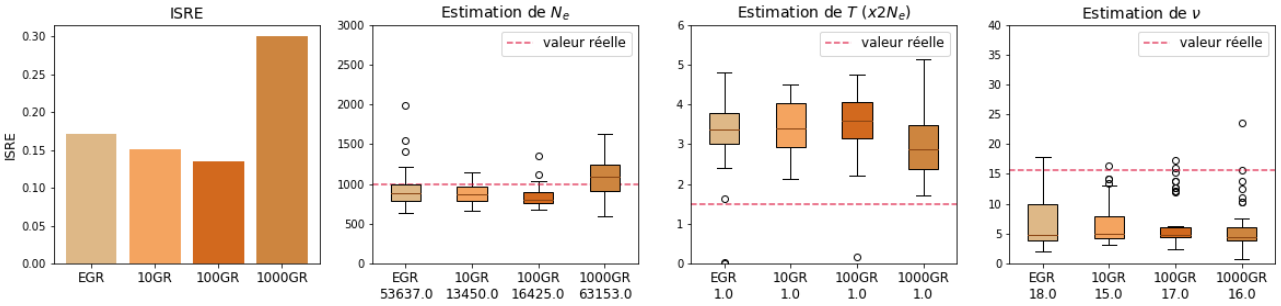

FIGURE 11 – **ISRE calculée et estimation des paramètres du modèle exponentiel**
EGR : recombinaison toutes les générations, 10GR : recombinaison toutes les 10 générations, 100GR : recombinaison toutes les 100 générations, 1000GR : recombinaison toutes les 1000 générations. Les chiffres correspondent à la variance.

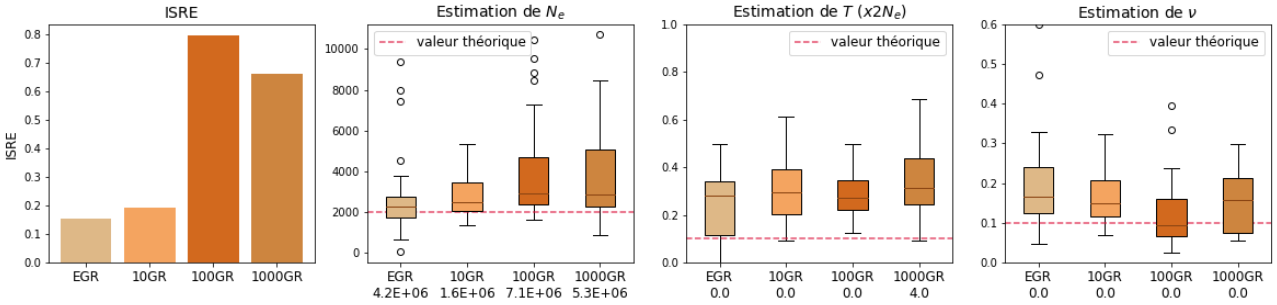

FIGURE 12 – **ISRE calculée et estimation des paramètres du modèle à 2 époques**
EGR : recombinaison toutes les générations, 10GR : recombinaison toutes les 10 générations, 100GR : recombinaison toutes les 100 générations, 1000GR : recombinaison toutes les 1000 générations. Les chiffres correspondent à la variance.

4 Discussion

4.1 Choix des hyper-paramètres et évaluation de l'inférence démographique

La première partie du stage et le choix des hyper-paramètres pour SMC++ a donc révélé que les histoires évolutives inférées par SMC++ sont très dépendantes des hyper-paramètres utilisés. Il est donc difficile de trouver des paramètres optimaux pour n'importe quel modèle, ce qui pose notamment problème lors de l'utilisation de SMC++ sur des données réelles.
Avec le jeu de paramètres retenu, nous avons donc inféré les histoires évolutives pour les différents modèles théoriques présentés. Le premier point que nous avons noté est que SMC++ n'infère pas correctement les temps récents (moins de 1000 générations, sachant qu'une génération correspond à environ 1h chez la levure). Ce point a été observé dans d'autres publications également[18]. En effet, il est souvent difficile d'estimer les temps récents d'une histoire évolutive en particulier lorsque l'on a un échantillon de petite taille par rapport à la taille de la population puisqu'il y a moins d'évènements de coalescence. Dans les courbes obtenues, on observe donc un plateau dans les temps récents qui n'est pas forcément attendu. Ce point a donc fortement influencé l'estimation des paramètres dans la suite.

En effet, nous notons par exemple pour le modèle exponentiel que N_e, la taille efficace de la population avant l'expansion, est relativement bien estimé (N_e caractérisant la taille pour toute la période ancienne), mais que ν, le ratio des tailles avant et après expansion, est très fortement sous-estimé. Ceci est dû à la difficulté d'inférer l'histoire très récente et donc la mau-

vaise inférence du $N_{e,final}$. Celui-ci est en effet sous-estimé ce qui induit une sous-estimation de ν. Mesurer le taux de croissance exponentielle sur la moitié de la période d'expansion serait donc plus judicieux, puisqu'on ne tiendrait pas compte des temps récents mal inférés.
Nous retrouvons un problème similaire avec l'estimation de T pour le modèle à 2 époques. Celui-ci se situe théoriquement proche de 200 générations, correspondant clairement dans des temps récents. SMC++ n'a donc pas réussi à l'inférer correctement et l'a sur-estimé (plaçant le changement de taille dans des temps plus anciens). Ces points confirment la mauvaise inférence des temps récents avec SMC++.
Il faut également noter que nous avons dans nos simulations un log-ratio $log(\frac{\rho}{\mu}) = -38.047$ ce qui est relativement faible. Un log-ratio faible est réputé difficile pour les méthodes d'inférence[14] (Figure S1), ce qui pourrait en partie expliquer les difficultés pour SMC++ d'inférer les paramètres.

4.2 Impact de la fréquence de recombinaison

SMC++ se basant sur un modèle théorique, si on lui impose une périodicité inhabituelle pour la recombinaison, nous violons les hypothèses de son modèle. Nous cherchons donc à étudier l'impact sur l'inférence et les limites du modèle.
Pour cela, nous avons estimé les paramètres (T, N_e, ν) pour différentes fréquences de recombinaison. Nous nous attendions à trouver soit que la fréquence de recombinaison n'a pas d'impact sur l'inférence (stabilité totale dans l'inférence des paramètres), soit qu'elle a un effet (comme montré avec *dadi* par Louis Ollivier en Annexe 3) qui se traduit par une augmentation de l'erreur lorsque la fréquence diminue et qui peut être due soit (1) à une augmentation de la variance des estimations, soit (2) à une augmentation du biais (sous ou sur-estimation) des paramètres.
Au vu des résultats obtenus, il est assez clair qu'il y a une augmentation de l'erreur lorsque la fréquence diminue (entre EGR et 1000GR), mais les fréquences intermédiaires ne suivent pas la même tendance. Il n'y a pas non plus d'augmentation notable du biais ou de la variance.
Ainsi, nous ne pouvons pas conclure concernant l'impact de la fréquence de recombinaison sur l'inférence démographique. Il semble assez clair que l'erreur pour 1000GR soit systématiquement plus élevée (pour les 3 modèles), et celle de 100GR également pour le modèle à 2 époques. IL serait judicieux de réaliser un test statistique pour voir si les différences de 1000GR (et 100GR pour le modèle 2 époques) sont significativement plus grandes. En effet, l'effet pourrait peut-être ne pas être linéaire comme attendu mais plutôt suivre un effet seuil avec une fréquence de recombinaison au dessus de laquelle le logiciel n'infère plus le modèle correctement.

4.3 Conclusion et perspectives

Le projet sur lequel j'ai travaillé s'intéresse à comprendre l'évolution de *S. cerevisiae* et notamment depuis sa domestication. Dans ce contexte, il est important d'établir si les logiciels d'inférence démographique sont capables de prendre en compte la particularité de la levure.
Indépendamment du processus de recombinaison, cette étude a tout d'abord mis en lumière une faiblesse de SMC++ : à savoir sa forte sensibilité au choix des hyperparamètres.
Concernant l'impact sur la fréquence de recombinaison, pour les trois modèles démographiques étudiés, nous n'observons pas de tendance claire, indiquant potentiellement que, dans le contexte démographique testé, la fréquence de recombinaison n'a pas d'effet particulier sauf pour 1000GR et 100GR dans le modèle à 2 époques, mais que SMC++ n'est pas robuste de manière générale.
Pour tirer une conclusion plus claire, il serait intéressant d'étendre la grille des paramètres à tester avec notamment la taille de l'échantillon n ou le temps à partir duquel SMC++ n'arrive pas à inférer l'histoire correctement. Il serait également intéressant de voir si *dadi* obtient une

tendance similaire pour 10GR et 100GR.

Références

[1] Hiren Karathia, Ester Vilaprinyo, Albert Sorribas, and Rui Alves. Saccharomyces cerevisiae as a Model Organism : A Comparative Study. *PLoS ONE*, 6(2) :e16015, February 2011.

[2] A. Goffeau, Massoud Ramezani-Rad, S. Rasmussen, A. Raynal, S. Rechmann, Miguel Remacha, José Revuelta, P. Rice, Guy-Franck Richard, P. Richterich, Michael Rieger, L. Rifken, L. Riles, Teresa Rinaldi, M. Rinke, A. Roberts, D. Roberts, Francesco Rodriguez, M. Rodríguez-Belmonte, and Marc Feuermann. The Yeast Genome Directory. *Nature*, 387, April 1997.

[3] Stacia R Engel, Fred S Dietrich, Dianna G Fisk, Gail Binkley, Rama Balakrishnan, Maria C Costanzo, Selina S Dwight, Benjamin C Hitz, Kalpana Karra, Robert S Nash, Shuai Weng, Edith D Wong, Paul Lloyd, Marek S Skrzypek, Stuart R Miyasato, Matt Simison, and J Michael Cherry. The Reference Genome Sequence of Saccharomyces cerevisiae : Then and Now. *G3 Genes—Genomes—Genetics*, 4(3) :389–398, 03 2014.

[4] Wang Qi-Ming Liu Wan-Qiu Shi Jun-Yan Li Kuan Zhang Xiao-Ling Bai Feng-Yan Duan Shou-FuHan, Pei-Jie. The origin and adaptive evolution of domesticated populations of yeast from far east asia.

[5] David E. MacHugh, Greger Larson, and Ludovic Orlando. Taming the Past : Ancient DNA and the Study of Animal Domestication. *Annual Review of Animal Biosciences*, 5(1) :329–351, February 2017.

[6] Wikimedia Foundation. Cycle cellulaire levure, Consulté en juin 2022.

[7] Brassage interchromosomique, Consulté en juin 20222.

[8] Gilles Fischer, Gianni Liti, and Bertrand Llorente. The budding yeast life cycle : More complex than anticipated ? *Yeast*, 38(1) :5–11, 2021.

[9] Benjamin C Haller and Philipp W Messer. SLiM 3 : Forward Genetic Simulations Beyond the Wright–Fisher Model. *Molecular Biology and Evolution*, 36(3) :632–637, 01 2019.

[10] Ryan N. Gutenkunst, Ryan D. Hernandez, Scott H. Williamson, and Carlos D. Bustamante. Inferring the Joint Demographic History of Multiple Populations from Multidimensional SNP Frequency Data. *PLoS Genetics*, 5(10) :e1000695, October 2009.

[11] Benjamin C. Haller, Jared Galloway, Jerome Kelleher, Philipp W. Messer, and Peter L. Ralph. Tree-sequence recording in SLiM opens new horizons for forward-time simulation of whole genomes. *Molecular Ecology Resources*, 19(2) :552–566, 2019. _eprint : https ://onlinelibrary.wiley.com/doi/pdf/10.1111/1755-0998.12968.

[12] Jerome Kelleher, Alison M Etheridge, and Gilean McVean. Efficient Coalescent Simulation and Genealogical Analysis for Large Sample Sizes. *PLOS Computational Biology*, 12(5) :e1004842, May 2016.

[13] Jerome Kelleher, Kevin R. Thornton, Jaime Ashander, and Peter L. Ralph. Efficient pedigree recording for fast population genetics simulation. *PLOS Computational Biology*, 14(11) :e1006581, 2018.

[14] Jonathan Terhorst, John A. Kamm, and Yun S. Song. Robust and scalable inference of population history from hundreds of unphased whole-genomes. *Nature genetics*, 49(2) :303–309, February 2017.

[15] Niklas Mather, Samuel M. Traves, and Simon Y. W. Ho. A practical introduction to sequentially Markovian coalescent methods for estimating demographic history from genomic data. *Ecology and Evolution*, 10(1) :579–589, December 2019.

[16] Sara Sheehan. Scalable Algorithms for Population Genomic Inference. Technical report, 2015.

[17] IFB. Clsuter de calcul ifb-core, Consulté en juin 2022.

[18] Austin H Patton, Mark J Margres, Amanda R Stahlke, Sarah Hendricks, Kevin Lewallen, Rodrigo K Hamede, Manuel Ruiz-Aravena, Oliver Ryder, Hamish I McCallum, Menna E Jones, Paul A Hohenlohe, and Andrew Storfer. Contemporary Demographic Reconstruction Methods Are Robust to Genome Assembly Quality : A Case Study in Tasmanian Devils. *Molecular Biology and Evolution*, 36(12) :2906–2921, 08 2019.

Annexe 1

Afin d'estimer les paramètres démographiques des modèles (comme $N_{e,.}$ ou T), nous estimons les paramètres d'une fonction ayant la forme voulue (par exemple composée d'une fonction constante au-dessus d'un seuil non prédéfini et d'une exponentielle au-dessous) de manière à optimiser le fit à la courbe reconstruite par SMC++. Ceci est fait pour chacun des réplicats. Pour cela, nous avons utilisé la fonction `curve_fit` du package Scipy. Par exemple, la figure A1 montre le fit obtenu pour le modèle exponentiel. Le fit permet également de combler la mauvaise inférence de SMC++ dans les temps récents. Avec une fonction de cette forme, on peut facilement estimer T (le seuil), $N_{e,initial}$ et $N_{e,final}$ (pour obtenir ν).

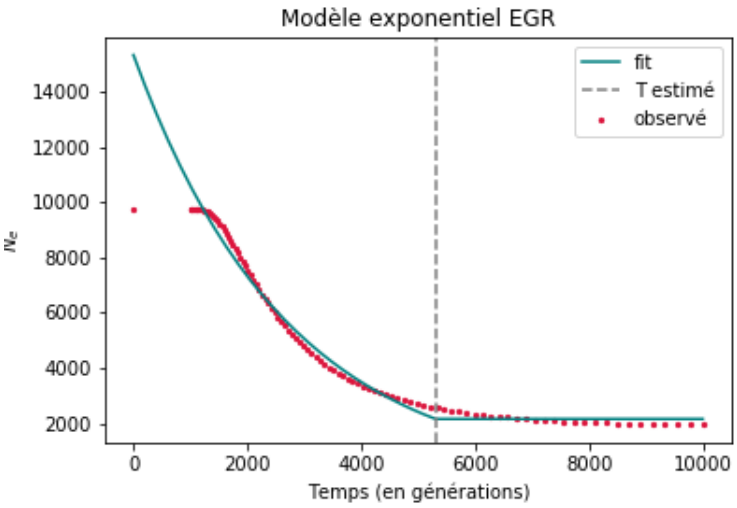

FIGURE A1 – **Fit obtenu pour un des réplicats du modèle exponentiel EGR. Observé : courbe inférée par SMC++.**

Cependant, pour le modèle à 2 époques, l'obtention du fit était trop dépendante des valeurs initiales données à la fonction curve_fit et nous n'avons pas réussi à obtenir un fit de la forme théorique (les données observées étaient assez éloignées de la courbe théorique). On a donc préféré ajuster aux données une sigmoïde (beaucoup plus proche), de la forme :

$$f(t, a, b, c) = \frac{c}{1 + ae^{-bt}}$$

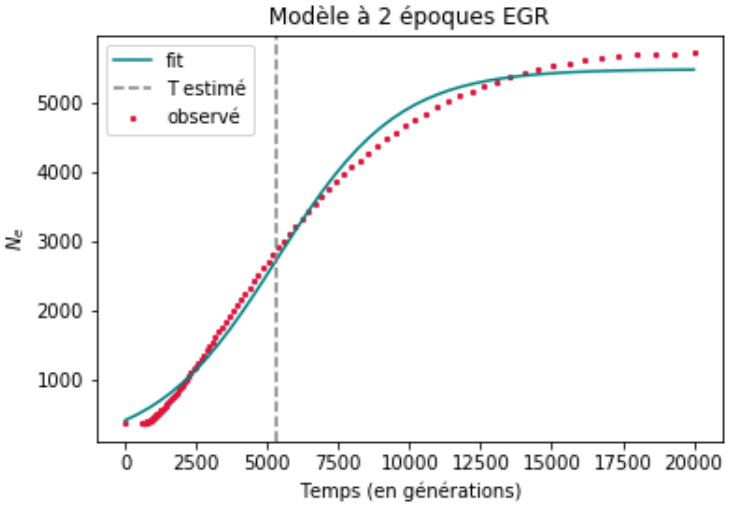

FIGURE A2 – **Fit obtenu pour un des réplicats du modèle exponentiel EGR**

On obtient, pour un réplicat, la courbe ci-contre (Fig **??**. Pour estimer T, on a ensuite cherché le point d'inflexion de la sigmoïde, c'est-à-dire le point pour lequel la dérivée est maximale. On peut également estimer les autres paramètres comme $N_{e,initial}$ et $N_{e,final}$ en calculant f aux points $t = 0$ et $t = 20\,000$.

Annexe 2

Pour observer la reconstruction d'une même histoire évolutive (pour le modèle exponentiel) avec différents paramètres de SMC++, on a calculé la courbe moyenne des différents réplicats. On observe ici très clairement que l'histoire évolutive inférée est très dépendante du jeu de paramètres choisis.
En effet, par exemple, pour certains jeux de paramètres (comme spline cubic, k10, rp1) on obtient une courbe presque constante et très éloignée de la courbe attendue.

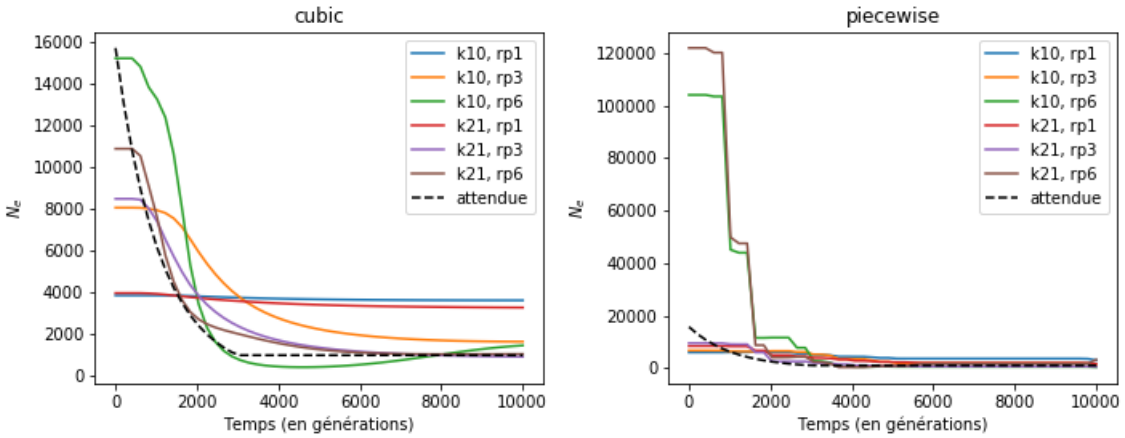

FIGURE A3 – **Effet des différents paramètres sur l'inférence démographique, modèle exponentiel**

Annexe 3

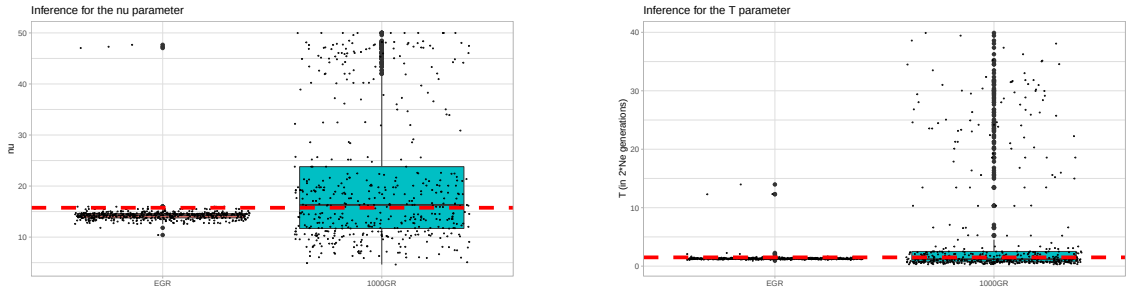

FIGURE A1 – **Estimation de T et ν pour le modèle exponentiel par *dadi***

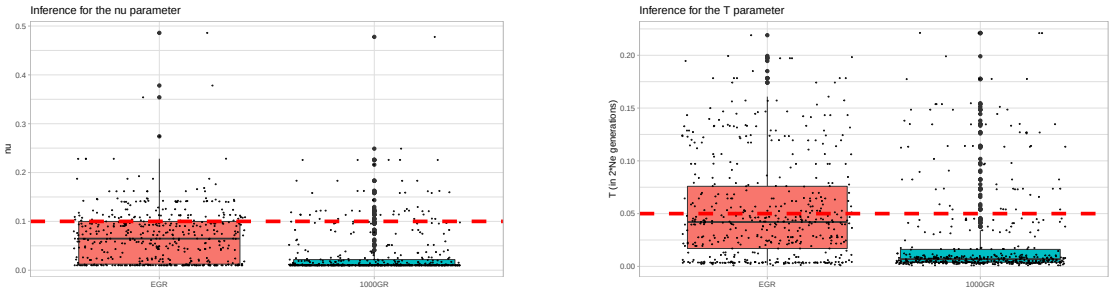

FIGURE A2 – **Estimation de T et ν pour le modèle à 2 époques par *dadi***