Guide d'utilisation du package R *autohaplo* pour l'identification automatisée d'haplotypes

Réunion du jeudi 5 octobre 2017

Marc-André Lemay
Aurélie Tardivel

Principe de la méthode

- Identification de marqueurs en déséquilibre de liaison (*linkage disequilibrium* ou LD) de part et d'autre d'un gène d'intérêt.
- La méthode repose sur l'idée que des marqueurs en LD de part et d'autre d'un gène devraient être en LD avec des sites polymorphiques dans le gène.
- On espère ainsi que les marqueurs conservés au terme de l'analyse définissent des haplotypes ayant potentiellement un intérêt fonctionnel.

Étapes de la méthode

- Inputs : jeu de données génotypiques (SNPs) et position centrale d'un gène d'intérêt
- Filtration (optionnelle) du jeu de données
- Calcul du LD et *clustering* : les marqueurs en LD d'un même côté du gène sont regroupés pour éviter la redondance d'information.
- Sélection des marqueurs : seules les paires de marqueurs en LD de part et d'autre du gène sont conservées pour définir les haplotypes.

Caractéristiques du package

- Le package accepte présentement des données en format VCF et hapmap.
- Le LD peut être calculé en tenant compte ou non de la structure et/ou du kinship de la population.
- À des fins de performance, il est possible d'utiliser des mesures de LD différentes lors des étapes de *clustering* et de sélection des marqueurs (e.g. r^2 seul pour le clustering, puis r^2_{vs} pour la sélection).

Installation du package

- Installation des packages disponibles sur CRAN :
  - `install.packages(c("ggplot2", "LDcorSV", "Matrix", "reshape2", "stringr", "devtools"))`
- Installation des packages disponibles sur Bioconductor :
  - `source("https://bioconductor.org/biocLite.R")`
  - `biocLite()`
  - `biocLite(c("GenomeInfoDb", "snpStats", "SummarizedExperiment", "VariantAnnotation"))`
- Installation du package autohaplo :
  - `setwd("autohaplo") # répertoire contenant le code du package`
  - `devtools::install()`
- Le package est maintenant installé sur votre machine. Pour l'utiliser, il faut appeler `library(autohaplo)` comme vous feriez pour tout autre package.

Utilisation du package : paramètres

- Le package comprend une fonction nommée `haplo_params` qui sert spécifiquement à construire les paramètres de l'analyse.
- Chacun des paramètres est réglé par un argument de cette fonction.
- Les paramètres se regroupent en 3 catégories :
  - les inputs
  - les paramètres de filtration
  - les paramètres relatifs au calcul du LD et à la sélection des marqueurs.

Paramètres : fichiers input

- `input_file` (hapmap ou vcf) : nom du fichier de génotypes
- `kinship_file` : nom du fichier de kinship
- `structure_file` : nom du fichier de structure
- `chr_db_file` : nom du fichier décrivant les chromosomes de l'espèce d'intérêt
- `gene_db_file` : nom du fichier décrivant le ou les gènes d'intérêt
- `gene_name` : nom du gène d'intérêt

Exemple de `kinship_file`

	RE.048	RE.056	RE.064	RE.072
RE.048	2.33888	0.15377852	-0.004576839	-0.18073153
RE.056	0.15377852	2.2922938	-0.1446677	0.06119093
RE.064	-0.004576839	-0.1446677	1.8447183	-0.2753211
RE.072	-0.18073153	0.06119093	-0.2753211	1.8041039

Exemple de `gene_db_file`

gene	chr	start	end
E1	Chr06	20207077	20207940
Gia	Chr10	45294735	45316121
PhyA3	Chr19	47633059	47641958
TFL1	Chr19	45183357	45185175
T	Chr06	18731105	18738025
PhyA2	Chr20	33236018	33241692

Exemple de `structure_file`

	p1	p2	p3	p4	p5	p6			
RE.048	0.739295		0.000002		0.000002		0.000002	0.211583	0.049115
RE.056	0.350868		0.000002		0.649123		0.000002	0.000002	0.000002
RE.064	0.061704		0.000002		0.000002		0.938286	0.000002	0.000002
RE.072	0.000002		0.000002		0.357917		0.000002	0.642074	0.000002
RE.080	0.000002		0.000002		0.000002		0.999988	0.000002	0.000002
RE.088	0.745346		0.000002		0.254645		0.000002	0.000002	0.000002
RE.096	0.000002		0.000002		0.000002		0.914650	0.000002	0.085341
RE.049	0.389966		0.000002		0.000002		0.116922	0.153741	0.339367
RE.065	0.000002		0.000002		0.832563		0.167427	0.000002	0.000002
RE.073	0.000002		0.000002		0.000002		0.999988	0.000002	0.000002
RE.081	0.000002		0.252811		0.000002		0.000002	0.000002	0.747179
RE.089	0.083236		0.042553		0.874204		0.000002	0.000002	0.000002
RE.097	0.000002		0.000002		0.000002		0.999988	0.000002	0.000002
RE.058	0.643629		0.356362		0.000002		0.000002	0.000002	0.000002
RE.074	0.245948		0.132237		0.000002		0.000002	0.621808	0.000002
RE.082	0.000002		0.000002		0.000002		0.000002	0.448213	0.551778
RE.090	0.122485		0.000002		0.000002		0.877505	0.000002	0.000002
RE.098	0.000002		0.999988		0.000002		0.000002	0.000002	0.000002
RE.051	0.000002		0.182168		0.000002		0.207393	0.185577	0.424857
RE.059	0.569578		0.000002		0.000002		0.285625	0.144790	0.000002
RE.075	0.000002		0.999988		0.000002		0.000002	0.000002	0.000002
RE.083	0.000002		0.000002		0.000002		0.000002	0.000002	0.999988
RE.091	0.387641		0.015139		0.217400		0.167466	0.212352	0.000002
RE.099	0.000002		0.321132		0.000002		0.346955	0.331906	0.000002

Exemple de `chr_db_file`

chr	length
Chr01	56831624
Chr02	48577505
Chr03	45779781
Chr04	52389146
Chr05	42234498
Chr06	51416486
Chr07	44630646
Chr08	47837940
Chr09	50189764
Chr10	51566898
Chr11	34766867
Chr12	40091314
Chr13	45874162
Chr14	49042192
Chr15	51756343
Chr16	37887014
Chr17	41641366
Chr18	58018742
Chr19	50746916
Chr20	47904181

Paramètres : filtration des marqueurs

- `min_alt_threshold` : proportion minimale de l'allèle mineur
- `max_het_threshold` : proportion maximale d'hétérozygotes
- `max_missing_threshold` : proportion maximale de données manquantes
- `min_allele_count` : nombre minimal d'observations de l'allèle mineur
- `max_marker_to_gene_distance` : distance maximale entre le marqueur et le centre du gène

Paramètres : calcul du LD et sélection des marqueurs

- `R2_measure` : mesure de LD utilisée pour la sélection des marqueurs
  - `r2` : r^2 simple
  - `r2s` : r^2 corrigé pour la structure
  - `r2v` : r^2 corrigé pour le kinship
  - `r2vs` : r^2 corrigé pour la structure et le kinship
- `cluster_R2` : mesure de LD utilisée pour le regroupement des marqueurs (même que `R2_measure` par défaut)
- `cluster_threshold` : LD minimal pour que des marqueurs (d'un même côté du gène) soient regroupés
- `marker_independence_threshold` : LD minimal pour que deux marqueurs de part et d'autre du gène soient considérés en LD
- `max_flanking_pair_distance` : distance maximale entre deux marqueurs en LD de part et d'autre du gène

Calcul des haplotypes et visualisation de l'output

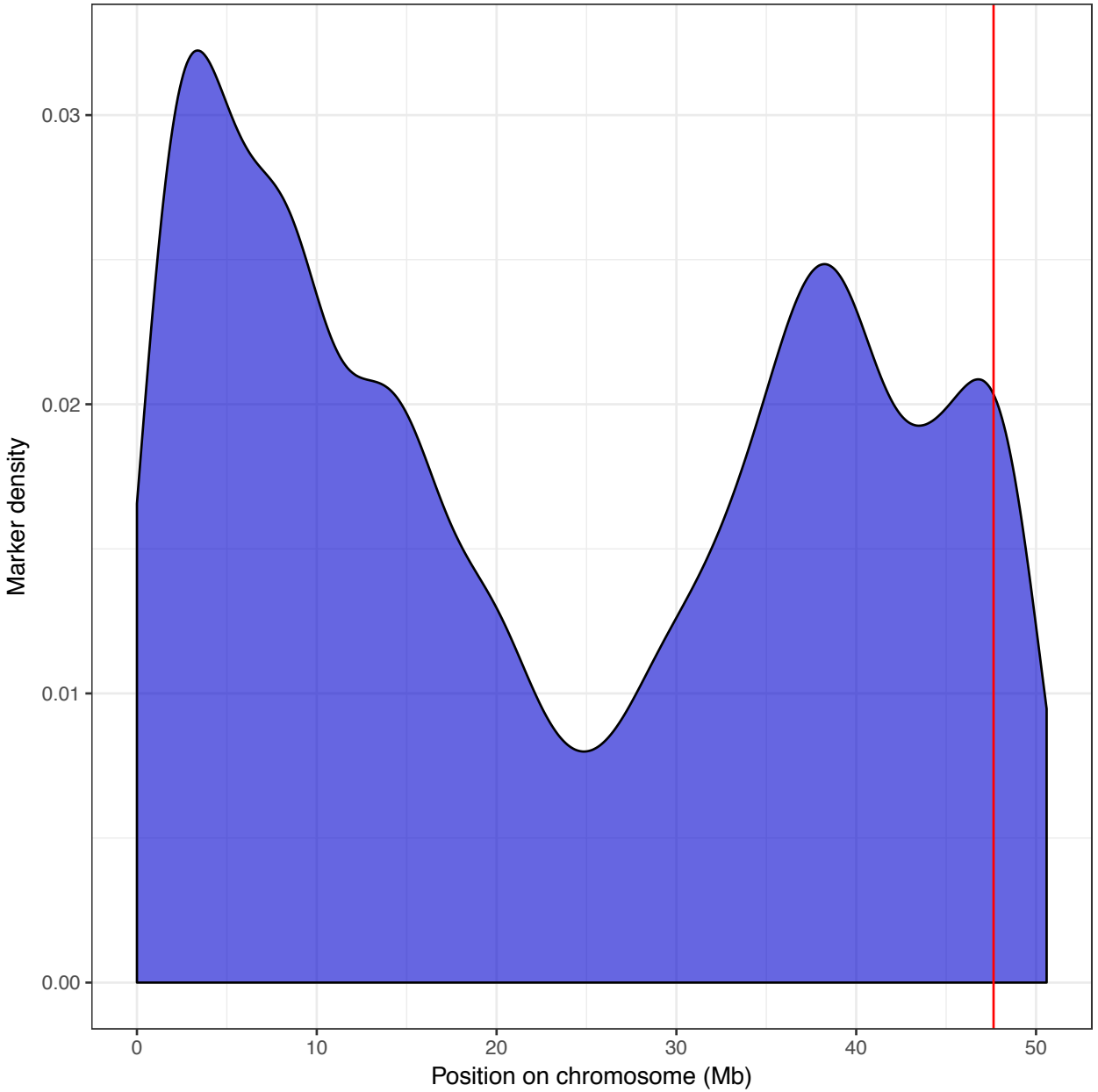
- L'analyse est lancée en fournissant à la fonction `haplo_selection` l'objet contenant les paramètres.

- L'objet généré par cette fonction est soumis à la fonction auto`haplo_output` afin de générer les graphiques et les fichiers de marqueurs résultant de l'analyse (la liste des fichiers générés peut être personnalisée).

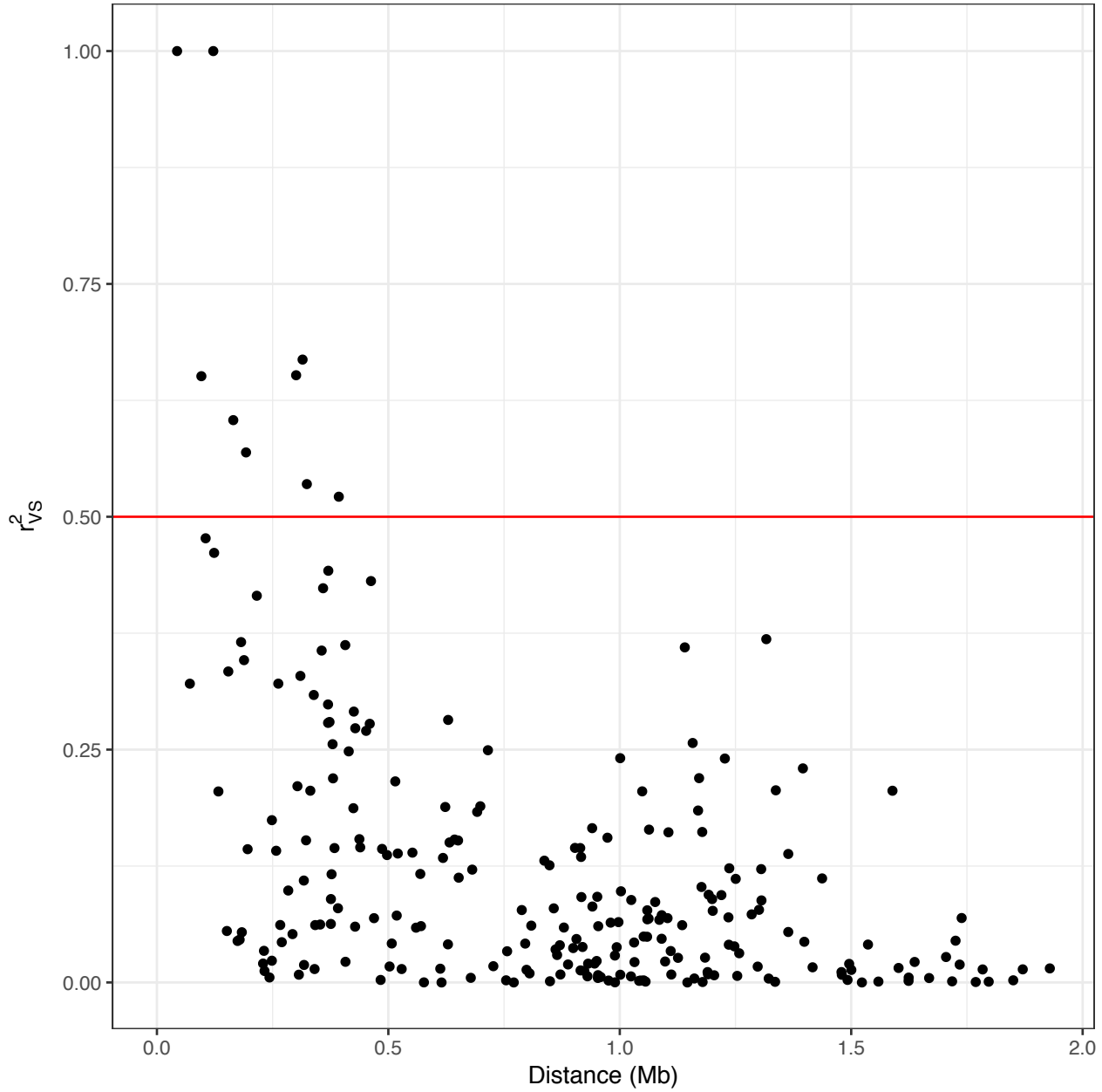
Utilisation du package : fichiers output

- Le package permet de générer cinq types de fichiers :
  - Densité des marqueurs le long du chromosome
  - Graphique du LD en fonction de la distance entre les marqueurs
  - Matrice du LD entre les marqueurs
  - Représentation visuelle des génotypes pour différents marqueurs
  - Fichiers hapmap ou vcf des marqueurs conservés
- À cinq étapes différentes de l'analyse :
  - Avant filtration
  - Après filtration
  - Après regroupement (*clustering*) des marqueurs
  - Après sélection des marqueurs (marqueurs regroupés ou non)
  - Après la détermination des haplotypes
- Attention : ce ne sont pas toutes les combinaisons qui ont un intérêt ou même un sens.

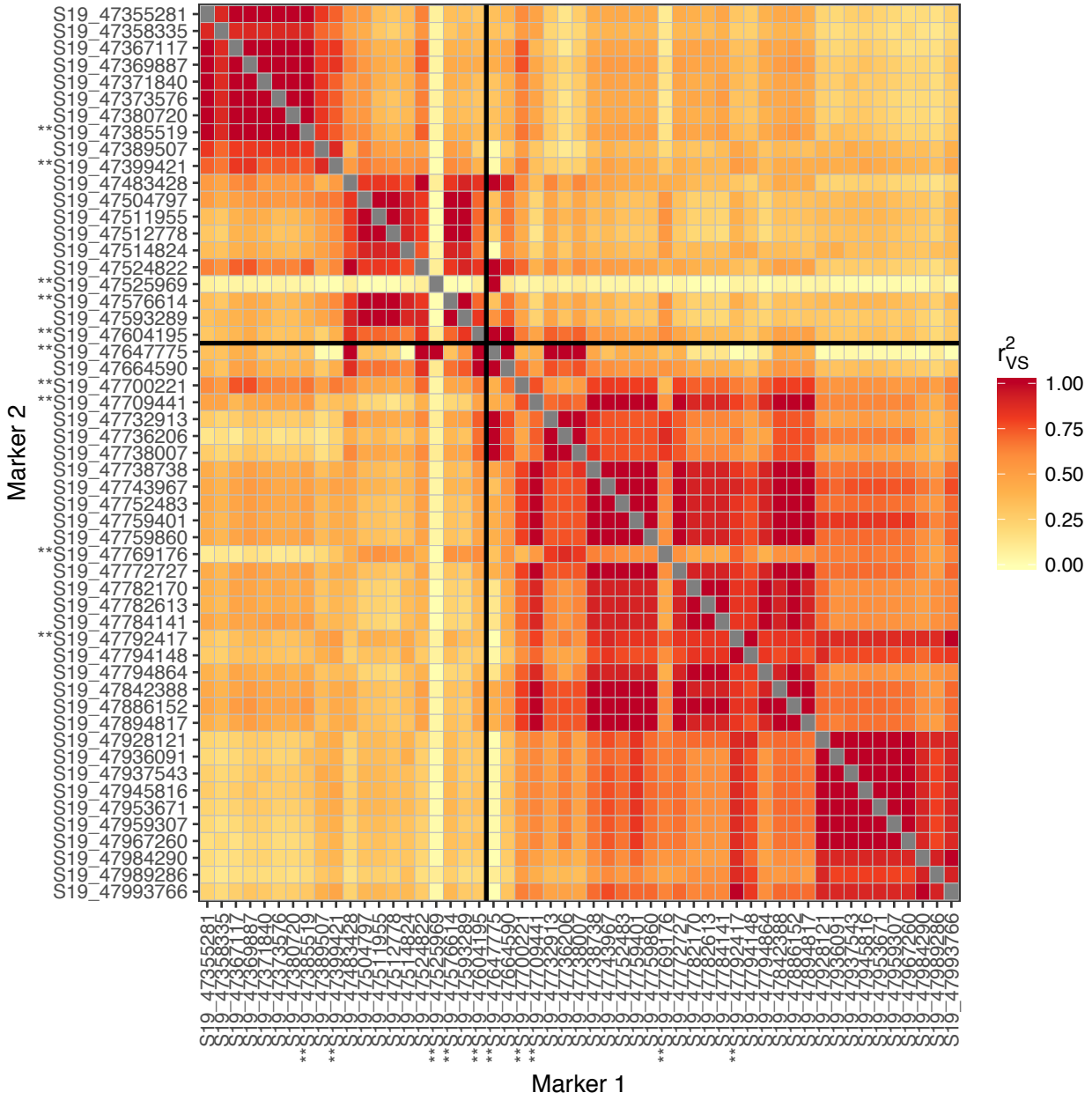
Graphique de densité des marqueurs



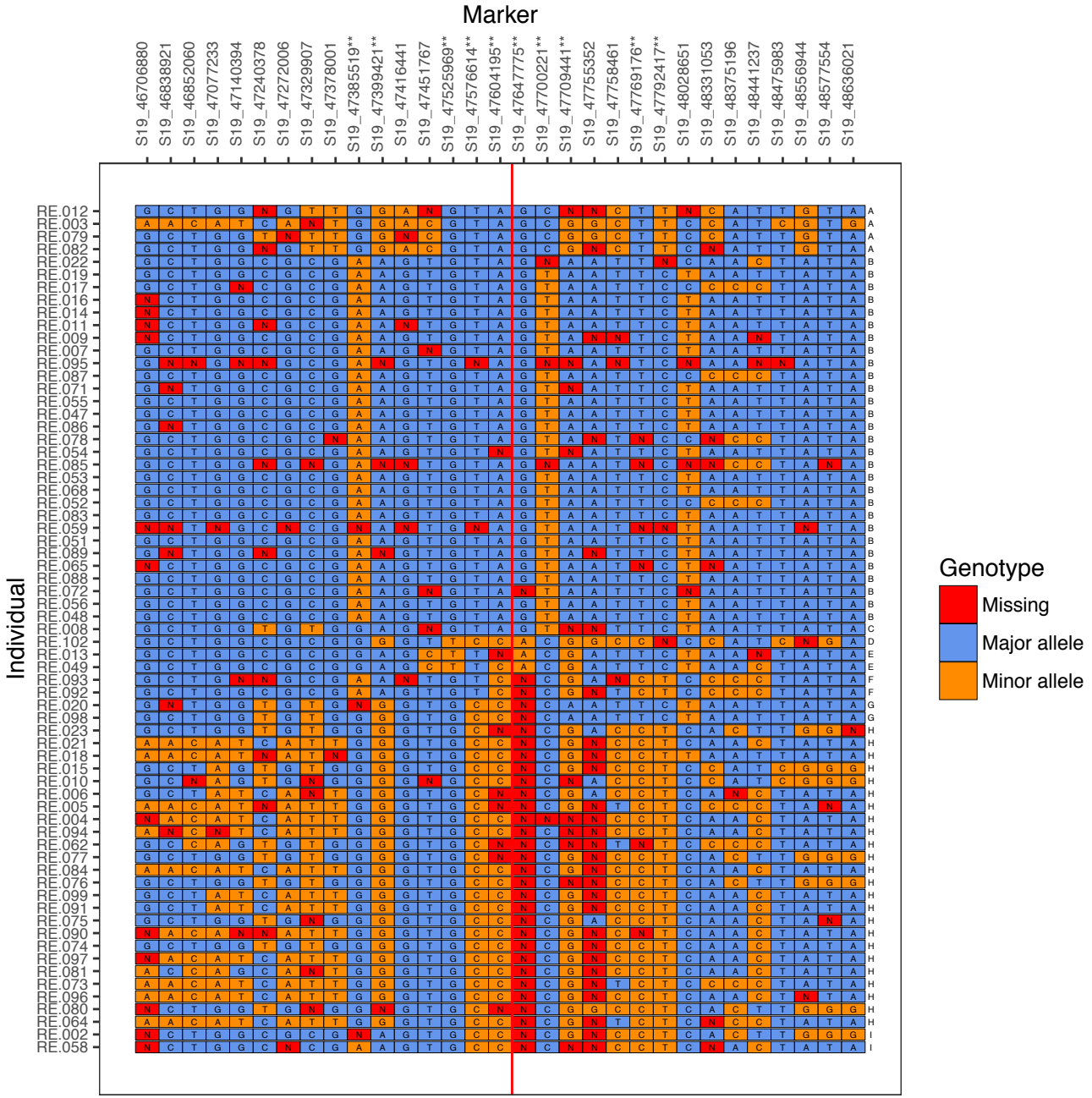
LD en fonction de la distance



Matrice de LD



Génotypes à différents marqueurs



Haplotypes par individu