НИС АДНЖ · МЕТОД ЛОКАЛЬНОГО ПРОИСХОЖДЕНИЯ

Loter

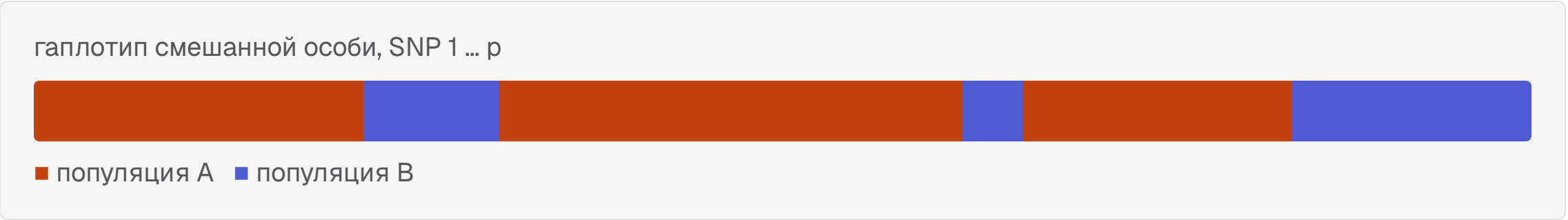
Локальное происхождение как задача оптимизации: несовпадения + λ · переключения

Dias-Alves, Mairal, Blum. *Loter: A Software Package to Infer Local Ancestry for a Wide Range of Species.* Mol Biol Evol 35(9):2318–2326, 2018

Максим Кирилюк · 19 ноября 2026

ЗАДАЧА

LAI — раскрасить хромосому по происхождению



- **Вход:** фазированный гаплотип *h* смешанной особи и референсные панели K популяций.
- **Выход:** для каждого SNP *j* — метка происхождения $a_j \in \{1, \ldots, K\}$.
- Это **одномерная сегментация**: метка на каждой позиции, и метки идут длинными кусками.

Постановка: Dias-Alves et al. 2018, «New Approaches»

Длина тракта — это часы

Каждое поколение рекомбинация режет хромосому. В модели симуляций статьи длина куска в Морганах ~ Exp(μ), μ — поколений после смешения: **среднее 1/μ M = 100/μ cM**.

поколений после смешения μ = **10**



μ = **10** поколений → точек смены на 100 cM: **14** (ожидание ≈ 10). Средний кусок: теория **100/μ = 10 cM**, здесь 6.7 cM.

Чем старше смешение, тем короче куски — и тем труднее их увидеть. Отсюда вся дальнейшая математика.

Simulations: «length (measured in Morgans) of an ancestry chunk is drawn according to an exponential distribution of rate μ». Соседние одноцветные куски сливаются в один тракт.

Loter почти ничего не просит

	HAPMIX	LAMP-LD	RFMix	Loter
Фазированные гаплотипы смешанных	нет	нет	да	да
Число предковых популяций	2	2, 3, 5	≥ 2	**≥ 2**
Генетическая карта (cM)	да	нет (физ. позиции)	да	**нет (только порядок SNP)**
Лимит числа SNP	нет	50 000	нет	нет
Коррекция ошибок фазирования	нет	не нужна	да	да, при K = 2

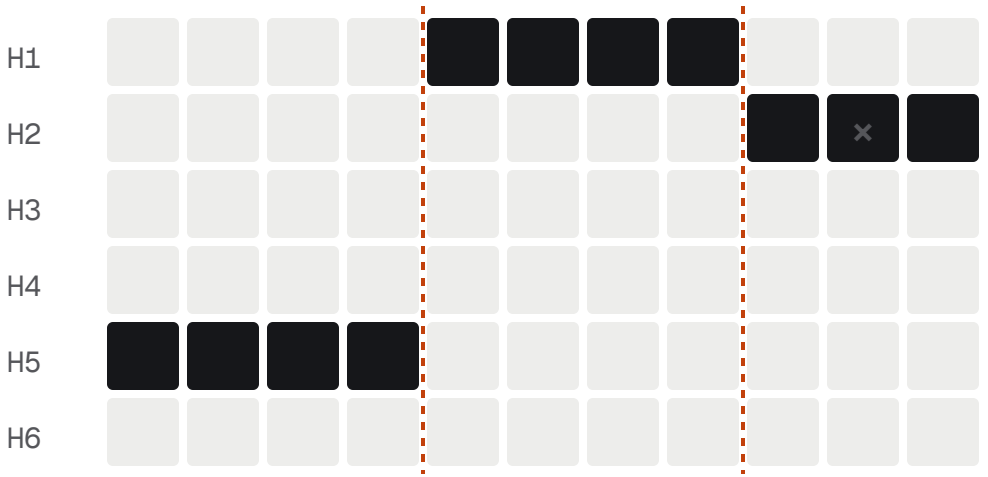
RFMix ещё просит число поколений и размер окна. У Loter — **один параметр λ**, и даже его в итоге усредняют.

Table 1 и раздел о параметрах RFMix, Dias-Alves et al. 2018

Гаплотип — мозаика из референсных гаплотипов

- Панель: 2n фазированных гаплотипов H_1 … H_{2n}, $H_{ij} \in \{0, 1\}$; каждый принадлежит одной из K популяций.
- Путь $s = (s_1, \ldots, s_p) \in \{1, \ldots, 2n\}^p$: $s_j = k$ — на SNP j цель копирует H_k.
- Происхождение: a_j = популяция гаплотипа H_{s_j}.

Идея копирования — Li & Stephens (2003), как в HAPMIX. Но без вероятностей.



SNP 1 … 11; пунктир — переключения (2 × λ)

Пример рис. 2: s = (5,5,5,5,1,1,1,1,2,2,2), одно несовпадение (×), два переключения → цена **1 + 2λ**.

ЗАДАЧА ОПТИМИЗАЦИИ · EQ. (1)

Одна функция стоимости

$$C(s) = \sum_{j=1}^{p} \left| h_j - H_{s_j, j} \right| + \lambda \sum_{j=1}^{p-1} 1_{s_j \neq s_{j+1}}$$

Потеря — число SNP, где цель не совпала с копируемым гаплотипом.

Регуляризация — λ × число переключений донора. Большой λ → длинные куски.

- Ищем $\hat{s}$ = argmin C(s) по всем путям; ответ a_j = pop($\hat{s}_j$).
- Переключение A1 → A2 стоит λ, но **происхождение не меняет**: смена донора ≠ граница тракта.

Eq. (1), Dias-Alves et al. 2018

МАСШТАБ

Перебрать пути нельзя

$$\left|\{1, \ldots, 2n\}^p\right| = (2n)^p$$

Например, 2n = 200 референсных гаплотипов и p = 500 000 SNP: $\mathbf{200^{500\,000} \approx 10^{1\,150\,515}}$ путей.

Но стоимость складывается из **локальных** слагаемых: несовпадение зависит от одного SNP, штраф — от соседней пары. Значит, лучший путь до SNP j можно достраивать по одному шагу.

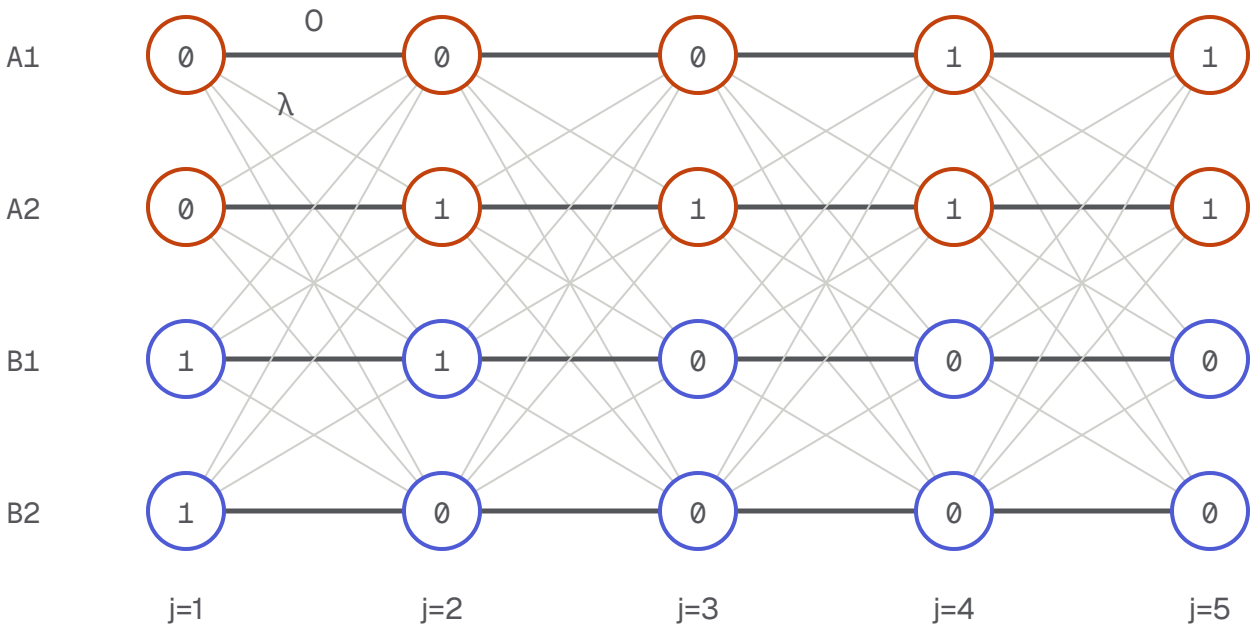
Если $s_p = s_{p-1}$: $C(s_{1..p}) = C(s_{1..p-1}) + |h_p - H_{s_p p}|$ (eq. 2)

Если $s_p \neq s_{p-1}$: $C(s_{1..p}) = C(s_{1..p-1}) + |h_p - H_{s_p p}| + \lambda$ (eq. 3)

p = 500 000 — размер Populus-набора в сравнении скорости; 500 000 · $\log_{10}200 \approx 1\,150\,515$ — наш расчёт. Eq. (2)–(3), Dias-Alves et al. 2018

ГРАФ · РИС. 3

Лучший путь — кратчайший путь в слоистом графе



- Вершина (i, j): «на SNP j копируем H_i», вес = $|h_j - H_{ij}|$.
- Ребро по строке — 0, ребро в другую строку — λ.
- Цена пути слева направо = C(s).
- Это алгоритм Витерби, где вместо −log вероятностей — стоимости.

Fig. 3, Dias-Alves et al. 2018. Числа в вершинах — первые 5 SNP учебного примера со следующих слайдов; сравнение с Витерби — наша формулировка.

Рекуррентность и трюк с минимумом

$$Q(i, j) = \left| h_j - H_{ij} \right| + \min(Q(i, j-1),\ m_{j-1} + \lambda)$$

$$m_{j-1} = \min_{i'} Q(i', j-1)$$

- Q(i, j) — цена лучшего пути по SNP 1..j, который на SNP j стоит в строке i; $Q(i, 1) = |h_1 - H_{i1}|$.
- Прыгать выгодно только с лучшей строки → m_{j-1} считаем **один раз на столбец**, клетка — O(1).
- Итог: **O(2n · p)** на целевой гаплотип. Ответ — $\min_i Q(i, p)$, путь — обратным проходом по указателям.
- Память: Q нужен по одному столбцу, но указатели — вся таблица 2n × p.

Eq. (4); в статье опечатка i′ ∈ {1..n} — гаплотипов 2n (сверено с кодом graph.cpp:156–163). Сложность в статье записана как O(n × p) — тот же порядок.

Заполняем Q по столбцам, потом идём назад

λ = **1** — пришли прыжком

SNP j	1	2	3	4	5	6	7	8	9	10
A1	0	0	1	0	0	0	1	0	0	0
A2	0	1	0	0	0	0	0	0	1	0
B1	1	1	1	1	1	0	1	1	1	1
B2	1	0	1	1	1	1	1	1	0	1
цель h	**0**	**0**	**1**	**1**	**1**	**0**	**1**	**0**	**1**	**0**

Q(A1, j)	0	0	0	1	2	2	2	2	3	3
Q(A2, j)	0	1	2	2	3	2	3	2	2	2
Q(B1, j)	1	2	1	1	1	1	1	2	2	3
Q(B2, j)	1	1	1	1	1	2	2	3	4	4
a_j	A	A	B	B	B	B	B	A	A	A

Готово. Путь: A1 A1 B1 B1 B1 B1 B1 A2 A2 A2
несовпадений 0 + λ · переключений 1 · 2 = **2**. Из 2 переключений смен происхождения: 2. Попробуй другой λ.

Шум и короткий тракт — для λ одно и то же

Внутри A-фона кусок, где цель совпадает с B на **d** информативных SNP. Два варианта: остаться в A — заплатить **d** несовпадений; уйти в B и вернуться — **2λ**.

вставка B выгоднее ⇔ **d > 2λ** наш вывод из eq. (1), одиночная вставка

λ = **2** сетка Loter: 1.5 … 5

цель h: тёмная клетка — SNP, где h совпадает с B, а не с A



ответ Loter при λ = 2: оранжевый — A, синий — B



d=1	d=2	d=3	d=4	d=6	d=8	d=11
стёрт	стёрт	стёрт	ничья	найден	найден	найден

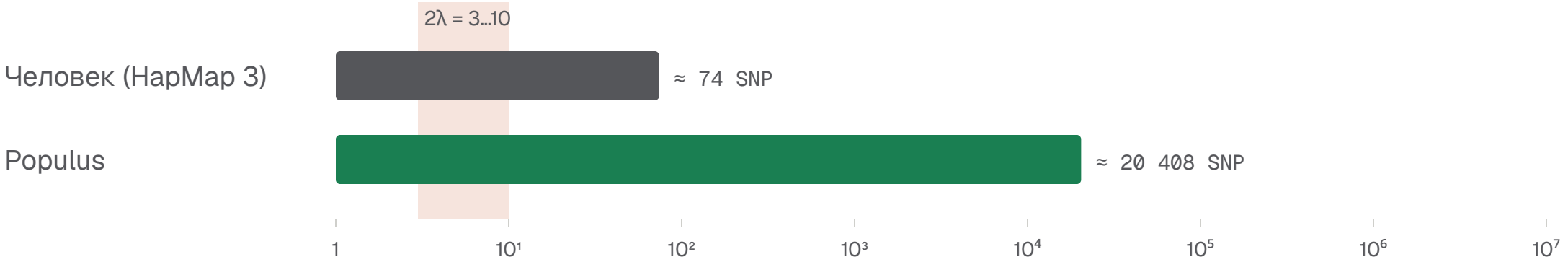
Порог 2λ = **4**: найдено 3 из 7 вставок. Шум d = 1 стёрт. Цена решения 22.

d — (несовпадений с A) − (несовпадений с B) внутри куска. Это сравнение двух путей, не общий порог: в общем случае ΔC = −d + λ·ΔS (у края хромосомы ΔS = 1, порог d > λ; близкие вставки DP может объединить). Виджет — настоящий DP на A = 0…0, B = 1…1.

Решает плотность SNP

При $\lambda \in [1.5, 5]$ вставке B внутри постоянного A нужно больше **3 … 10** (то есть ≥ 4 … 11) чистых информативных SNP. Средний SNP-шаг: человек **$2.7 \cdot 10^{-3}$ cM** (≈ 370 SNP/cM), Populus **$9.8 \cdot 10^{-6}$ cM** (≈ 100 000 SNP/cM).

$\mu =$ **500** SNP в среднем куске 100/μ cM, лог-шкала



μ = **500**: средний кусок 100/μ = **0.2 cM**. Это все SNP в куске; информативных (различающих популяции) — лишь часть.

Статья: LAI возможен и для смешения сотни поколений назад, «when SNP density is large enough, as for Populus data».

SNP-шаги — Discussion, Dias-Alves et al. 2018; перевод в SNP/cM и в SNP на кусок — наш расчёт.

Ансамбль: 8 значений λ × 20 бутстрепов

λ = 1 + 0.5 i, i = 1..8

1.5, 2, …, 5

×

20 бутстрепов

референсы каждой популяции — выборка с возвращением

→

160 решений

голосование большинством на каждом SNP

- Если у победителя меньше 75% голосов — происхождение берётся у соседних SNP (в коде CLI порог другой).
- Supp. Fig. S2: усреднение по λ лучше любого одного λ; бэггинг улучшает ещё. Лучший одиночный λ: 2 при μ = 400–500, иначе 5.
- Мысль: λ «сложно зависит» от мутаций и рекомбинации — поэтому его не подбирают, а усредняют.

«New Approaches» и Discussion, Dias-Alves et al. 2018; бутстреп по референсам — код local_ancestry.py (bcm-uga/Loter @ 5c91354).

Второе DP перераскладывает тракты между гаплотипами

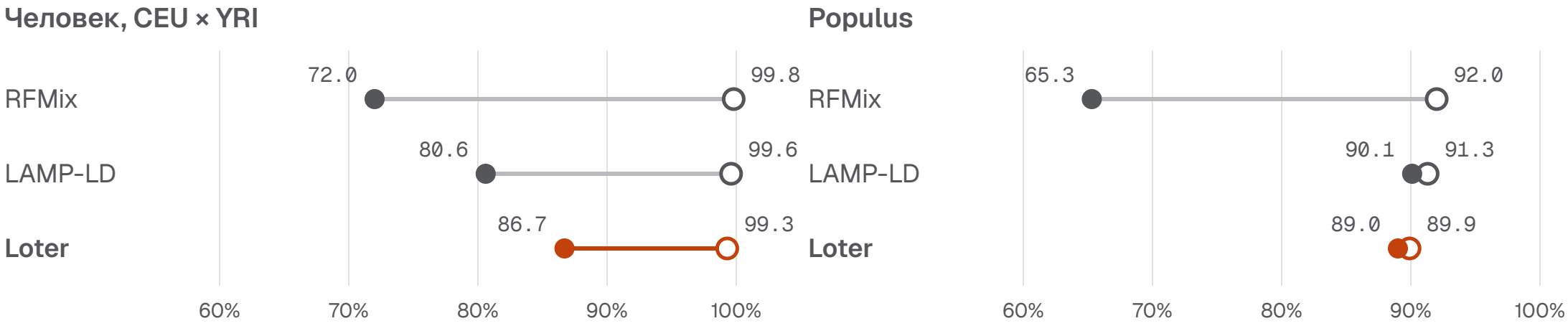
Только при K = 2. Складываем два гаплоидных ответа в диплоидный $d_j \in \{0, 1, 2\}$. Затем ищем две новые цепочки меток s, s′ (вспомогательные a, a′ $\in \{0,1\}^p$, **a + a′ = d**) с минимумом

$$C' = \sum_{j=1}^{p} \left| a_j - A_{s_j j} \right| + \sum_{j=1}^{p} \left| a'_j - A_{s'_j j} \right| + \lambda \cdot \mathrm{sw}(s) + \lambda \cdot \mathrm{sw}(s')$$

- sw — число переключений меток; $A_1 = (0,\ldots,0)$, $A_2 = (1,\ldots,1)$. Ответ — $x_j = A_{s_j j}$, $y_j = A_{s'_j j}$. Исключив a, a′, потеря на SNP равна $|d_j - x_j - y_j|$: DP торгует расхождением с d против переключений, x + y = d не гарантировано наш вывод. O(p) на особь.
- Проверка: 30% ошибок переключения фазы (CEU–YRI, 5 поколений) → та же диплоидная точность **97%**, что с истинными фазами.

Eq. (5) и «Accounting for Phase Errors», Dias-Alves et al. 2018 (во втором слагаемом в тексте статьи индекс s_j; по смыслу s'_j).

Loter почти не стареет вместе со смешением



○ μ = 5 поколений ● μ = 500 поколений

- Человек: при μ ≤ 20 RFMix и LAMP-LD лучше; при μ ≥ 150 лучший — Loter.
- Populus: Loter и LAMP-LD не зависят от возраста смешения; RFMix 92.0 → 65.3%.
- Скорость: 500 000 SNP, 20 тополей, 12 ядер — Loter **6 мин**, RFMix 28, LAMP-LD 58.

Figs. 4–5 и раздел о времени работы, Dias-Alves et al. 2018

Точность по SNP — не вся правда

медиана длины тракта, cM	истина	RFMix	Loter	LAMP-LD
Populus, 10 поколений	4.26	5.20	**2.31**	0.05
Человек, 10 поколений	9.03	9.99	**12.20**	9.02
Populus, 500 поколений	0.17	1.60	**0.33**	0.17

- Loter и LAMP-LD дают **ложные короткие куски** и режут длинные тракты на части (Populus, 10 поколений).
- RFMix при ≥ 200 поколениях восстанавливает тракты слишком длинными.

Fig. 6 и Supp. Fig. S6, Dias-Alves et al. 2018

Часы в деле: тополя

- 36 гибридов *P. trichocarpa* × *P. balsamifera*: в среднем 87% и 13% происхождения.
- Медиана тракта *balsamifera* — **0.76 cM** (квартили 0.25 и 1.47 cM).
- Симуляции на реальных генотипах с $\mu \in \{10, 50, 100, 200, 300, 500\}$: ближе всего распределение при **$\mu = 100$**.

Вывод статьи:
смешение было
≈ 100 поколений назад

Где модель упрощает

- **Слепота к расстояниям.** Переключение стоит λ, будь между SNP 1 п.н. или 1 Мб (в коде есть необязательные веса, по умолчанию выключены).
- **Короткие тракты** теряются: в сравнении A → B → A с постоянным A вставка с перевесом < 2λ информативных SNP невыгодна (на пороге — ничья); на плотных данных бывают ложные короткие куски.
- **Нужны фазированные данные** и для цели, и для референсов.
- **Панель решает:** TSI/LWK, 1 → 100 гаплотипов на популяцию — точность 41% → 98%.
- **λ без биологического смысла:** зависит от мутаций и рекомбинации «сложным образом» — отсюда ансамбль.

Results и Discussion, Dias-Alves et al. 2018; веса — код bcm-uga/Loter @ 5c91354

ИТОГ

Три мысли

- **LAI = одномерная сегментация.** Loter минимизирует несовпадения + λ · переключения донора.
- **Динамическое программирование** находит точный минимум за O(2n · p): внутренний минимум по строкам считается один раз на столбец.
- **λ задаёт разрешение:** вставка B внутри постоянного A выживает при d > 2λ. Поэтому решает плотность SNP, а выбор λ снимает ансамбль из 160 решений.

Спасибо! Вопросы?