LDJump: Estimating Variable Recombination Rates from Population Genetic Data

Philipp Hermann[1], Angelika Heissl[2], Irene Tiemann-Boege[2], and Andreas Futschik*[1]

[1]Department of Applied Statistics, Johannes Kepler University Linz, Austria.
[2]Institute of Biophysics, Johannes Kepler University Linz, Austria.

September 20, 2017

Abstract

Recombination results in the reciprocal exchange of genetic information occurring in meiosis which increases genetic variation by producing new haplotypes. Recombination rates are heterogeneous between species and also along different genomic regions. Large fractions of recombination events are often concentrated on short segments known as recombination hotspots. In this work we statistically inferred heterogeneous recombination rates by using relevant summary statistics as explanatory variables in a regression model. We used for this purpose a frequentist segmentation algorithm with type I error control to estimate the variation in local recombination rates. Under various simulation setups we have obtained very fast and accurate estimates. We also show an example of an inference on a 103kb region of the human genome and compare our inferred historical- and population-specific hotspots with results from experimental data (sperm-typing and double strand break maps). For the analyzed region, our method shows a good congruence between historical and experimental hotspots, except for one hotspot hypothesized to be population-specific. This method is implemented in the `R`-package *LDJump*, which is available from https://github.com/PhHermann/LDJump.

1 Introduction

Recombination is a necessary process during meiosis starting with the formation of DNA double-strand breaks (DSBs) that results in an exchange of genetic material between homologous chromosomes [Baudat et al., 2013]. This causes the formation of new haplotypes and increases

*Corresponding Author: andreas.futschik@jku.at

the genetic variability of populations. In population genetics, the recombination rate ρ is defined as $\rho = 4N_e r$, where N_e is the effective population size and r the recombination rate per base pair (bp) and generation. Recombination rates vary between species (human vs. chimpanzee see [Auton et al., 2012] or mice see [Smagulova et al., 2011]), populations within species (human populations like Africans and Europeans see [The 1000 Genomes Project Consortium, 2015, Pratto et al., 2014, Berg et al., 2010]), individuals within species (humans see [Pratto et al., 2014]), individuals of different sexes (see [Kong et al., 2010]), as well as along the genome with hot and cold regions of recombination see [Jeffreys et al., 2001, McVean et al., 2004, Myers et al., 2005]. Large fractions of recombination events are concentrated on short segments which are called hotspots (reviewed in [Arnheim et al., 2007]). The literature suggests to define human hotspots as showing an at least five-fold increase in rate compared to background recombination for a length of up to 2kb [McVean et al., 2004].

Molecular and evolutionary mechanisms of the process of recombination can be better understood with accurate estimates of the recombination rate in different regions of the genome [McVean et al., 2004, Chan et al., 2012]. Moreover, precise knowledge of the recombination rate variation along the DNA sequence improves inference from polymorphism data about e.g. positive selection [Sabeti et al., 2006], linkage disequilibrium [Hill and Robertson, 1968], and facilitates an efficient design and analyses of disease association studies [McVean et al., 2004].

Different approaches have been used to estimate recombination rates in humans differing in their genome-wide coverage, resolution, and active recombination. Experimental measures include whole genome sequencing or SNP typing of pedigrees of at least 2-3 generations [Coop et al., 2008, Kong et al., 2010, Halldorsson et al., 2016] which do not have a resolution below tens of the kilobases given that not many recombination events are captured. Direct measurements in sperm provide high resolution events at the level of a few hundreds of base pairs, but lack genome-wide coverage [Arnheim et al., 2007]. Finally, recombination has been inferred by the analyses of patterns of linkage disequilibrium. These represent historical, genome-wide recombination events and require the characterization of polymorphisms in several individuals within a population. Estimating the recombination rate (ρ) from patterns of linkage disequilibrium is difficult because recombination events have a low frequency and do not always leave traces in the genomic DNA sequences. One of the first approaches to estimate ρ was to compute a lower bound on the number of recombination events [Hudson and Kaplan, 1985, Wiuf, 2002, Myers and Griffiths, 2003]. Other methods to estimate ρ calculate moments or summary statistics [Hudson, 1987, Batorsky et al., 2011]. In [Wall, 2000, Wall, 2004], suitably chosen summary statistics such as the number of haplotypes (*haps*) are used. The author performs simulations with given *haps*, calculates the likelihoods for a series of ρ values, and chooses the value of ρ with the highest likelihood as estimator of the recombination rate.

Other methods estimate ρ via maximum likelihood [Kuhner et al., 2000, Fearnhead and Donnelly, 2001] or approximations to the likelihood [Hudson, 2001, Fearnhead and Donnelly, 2002, McVean et al., 2002, Li and Stephens, 2003, Wall, 2004]. The former methods rely on simulations using importance sampling [Fearnhead and Donnelly, 2001] or Markov chain Monte Carlo (MCMC) methods [Kuhner et al., 2000] to become computationally feasible. The latter approaches use a composite likelihood as in [Hudson, 2001], or a modified composite likelihood

as in [McVean et al., 2002]. Software implementations such as *LDhat* [McVean et al., 2004, Auton and McVean, 2007] and *LDhelmet* [Chan et al., 2012] are also available. [Kamm et al., 2016] extend this approach to account for demographic effects in their software package *LDpop*. Generally, computing approximate likelihoods requires a somewhat smaller computational effort than full likelihoods at the price of a slight loss in accuracy. An improvement of composite likelihood estimators via optimizing the trade-off between bias and variance has been proposed by [Gärtner and Futschik, 2016]. For a more technical discussion on composite likelihood in general see [Varin et al., 2011, Reid, 2013].
Recently, alternative fast estimates of ρ have been proposed by [Lin et al., 2013, Gao et al., 2016] that rely on regression. The software implementation is called *FastEPRR* and provides reliable estimates for samples of at least 50 individuals. This software estimates variable recombination rates in sliding windows. The authors note that they simultaneously minimize the prediction error and maximize Shannon entropy in information theory [Shannon, 1948]. In contrast to [Gao et al., 2016] our approach is also designed to work with small sample sizes.
In this paper we estimate variable recombination rates by partitioning the DNA sequence into homologous regions with respect to the recombination rates. Specifically, the DNA sequence is divided into small subregions (segments) and the recombination rate per segment is estimated via a regression based on summary statistics. A frequentist segmentation algorithm [Frick et al., 2014] is then applied to the estimated rates to obtain change-points in recombination. The algorithm controls the type I error and provides confidence bands for the estimator. [Futschik et al., 2014] use a similar approach to partition DNA sequences into homologous segments with respect to GC content. Section 2 contains a detailed description of our method called *LDJump*. In section 3 we assess *LDJump* and compare it with the popular software packages *LDhat* and *LDhelmet*. We provide an example of a prediction of a well-characterized region of the human genome of several populations in section 4 and summarize our findings in section 5. Further details on the regression model, bias correction, and more detailed comparisons are provided in the appendix.

2 Methods

Our approach consists of steps. First, we train a regression model to estimate a broad range of constant recombination rates. Subsequently, we apply a segmentation algorithm to estimate breakpoints in recombination rates subject to under type I error control against over-estimating the number of identified segments.

2.1 Regression Model for Constant Recombination Rates

Based on a set of summary statistics $\mathcal{X}$ (see Table 1) we fit the regression model (1) to estimate constant recombination within short DNA segments. More specifically, we use generalized additive models (GAM) [Wood, 2011] that estimate cubic spline functions $f_j(z_j)$ for covariates $z_j, j = 1, \ldots, m$ and linear (or quadratic) effects for covariates $x_k, k = 1, \ldots, l$. The structure

Variable	Description	Computation
*hats**	Constant recombination rate estimator of a segment	`pairwise` of *LDhat* [McVean et al., 2004]
*vapw**	Variance of the average pairwise differences per base pair	`convert` of *LDhat*[McVean et al., 2004]
*wath**	Wattersons's θ per base pair	`convert` of *LDhat* [McVean et al., 2004]
*apwd**	Average number of pairwise differences per base pair	`convert` of *LDhat* [McVean et al., 2004]
*fgts**	The number of pairs of sites for which the FGT indicates a recombination event per base pair	self implementation
*hahe**	Mean of haplotype heterozygosity for each pair of sites	`Hs` of `adegenet` [Jombart, 2008]
*rsqu**	Mean of r^2 for each pair of sites	`diseq` of `genetics` [Warnes et al., 2013]
*ldpr**	Mean of LD' for each pair of sites	`diseq` of `genetics` [Warnes et al., 2013]
haps	The number of haplotypes per base pair and per individual	`find_confs` of *LDhelmet* [Chan et al., 2012]
gcco	GC content: ratio of guanine and cytosine in the DNA sequence	`gc.content` of `ape` [Paradis et al., 2004]

Table 1: Summary statistics used in the regression model. Variables marked with an asterisk (*) had a significant effect.

of our GAM is

$$t(\rho_i) = f_1(z_{i1}) + \cdots + f_q(z_{iq}) + \beta_0 + \beta_1 x_{i1} + \cdots + \beta_k x_{ik} + \varepsilon_i, \tag{1}$$

for $i = 1, \ldots, n$. For a more detailed description of the regression model as well as the selection of explanatory variables see appendix A.1. Initial computations revealed variance heterogeneity. Hence, we transformed the population recombination rate ρ using a Box-Cox transformation $t(\rho)$ [Box and Cox, 1964]. In appendix A.2, we describe the choice of the transformation parameters (see (2)) and explore the issue of heterogeneous variances.
Table 1 contains the summary statistics of our set $\mathcal{X}$. We have rescaled some of the summary statistics by taking into account the length of the sequence or the number of individuals in the sample for which the summary statistics were computed. Since the (constant) recombination rate estimates of *LDhat* strongly rely on the number of haplotypes in the data, it is not surprising that the effect of the variable itself (*haps*) does not appear to be significant in our computations.

2.2 Segmentation Algorithm Estimating Variable Recombination Rates

[Frick et al., 2014] introduced a method for detecting change points in a function with observation errors distributed according to an exponential family. Their *simultaneous multiscale change-point estimator* (SMUCE) infers the number of change-points and their locations. The underlying function is also estimated and confidence bands are provided. Given the model assumptions are exactly satisfied, the probability of overestimating the number of change-points is controlled with a user specified type I error probability α. Slight deviations of the model assumptions will result in slightly overestimating the true number of segments. We use SMUCE for normally distributed errors (after transforming the response) to detect changes in the recombination rate. For more details on the algorithm see [Frick et al., 2014] and for a general overview on multiple change-point detection see [Niu et al., 2016].
In the first step *LDJump* divides the DNA sequence into a typically large number k of short segments. Summary statistics are computed separately for each segment and used in our regression model to estimate a local transformed recombination rate. The back-transformed

rates (natural scale of ρ, see appendix A.2) are used as input for the change point estimator. The estimated breakpoints reveal the heterogeneity of segments with respect to the recombination rates.

3 Simulations

We used the software package *scrm* of [Staab et al., 2014] to simulate populations with variable recombination rates and converted its output to *fasta*-files with the software package *ms2dna* of [Haubold and Pfaffelhuber, 2013]. In this section we compare *LDJump* with *LDhat*, the newer version *LDhat2*, as well as *LDhelmet* for constant and variable recombination rates, respectively. The runtime comparison is based on one core of an Intel Xeon E5-2630v3 2.4 1866, with 64GB DDR4-2133 RAM. Our analyses was performed in [R Core Team, 2017].

3.1 Constant Recombination Rate Estimation

We assess the quality of the regression model estimating constant recombination rates in comparison with the function `pairwise` of *LDhat* and `max_lk` of *LDhelmet*. We applied the functions following the default guidelines. Therefore, we simulated recombination rates between 0 and 0.1 with populations of sizes {10, 16, 20} and sequence lengths of {1000, 2000, 3000} base pairs. For each of these nine setups we simulated the same number of replicates (111 simulated $\rho \in [0, 0.1]$) yielding a sample size of almost 1000 observations.
Using the root mean squared error (RMSE $= \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{\rho}_i - \rho_i)^2}$) and the coefficient of determination R^2, we compare the accuracy of the mentioned methods. We visualize the estimators and the true values in Figure 1 along with a diagonal black line of a perfect fit. Both prediction measures show a better fit of the generalized additive model (purple plus signs: higher R^2 of 0.4974; smaller RMSE of 0.0256) compared with the software packages *LDhat* (red dots: 0.4447; 0.0290) and *LDhelmet* (green triangles: 0.2095; 0.0360).

3.2 Variable Recombination Rate Estimation

We compared and simulated two types of setup for variable recombination rate estimation: *simple* setups (sequences of length 10 and 20 kb with one hotspot) and *natural* setups (sequences of length 1Mb containing 15 hotspots). Both setups varied in background rates, sample sizes, hotspot intensities, and hotspot lengths. Furthermore we followed common practice and chose the recombination rate at hotspots as at least five times the background rate. Simulation results with *LDhat(2)* were computed as recommended with 10^6 iterations for the reversible-jump MCMC procedure, sampled every 4000 iterations, a burn-in of 10^5, and different block penalties of 0, 5, and 50. We also followed the recommendations for the computations with *LDhelmet* using for instance a window size of 50 SNPs, and 11 Padè coefficients. We applied the implemented function `smuceR` within the R-package `stepR` [Hotz and Sieling, 2016] to estimate the change-points of the back-transformed regression based recombination rate estimates.

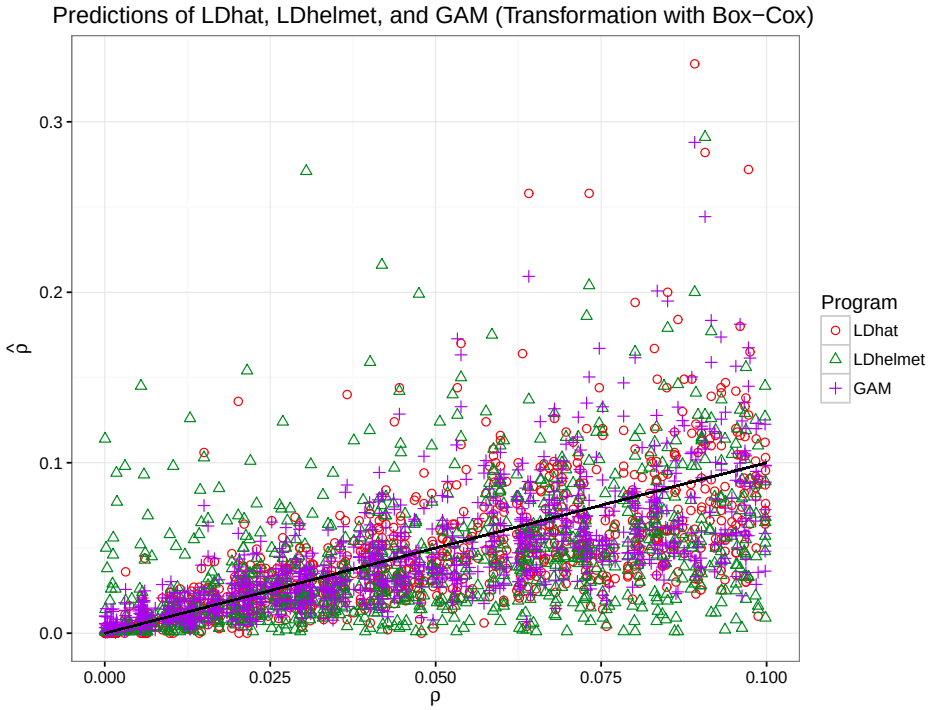

Figure 1: Predicted constant recombination rates versus their true values estimated with *LDhat* (red dots), *LDhelmet* (green triangles) and the GAM (purple plus signs). The black diagonal line shows the perfect fit.

3.2.1 Simple Setups

We simulated populations of sizes {10, 16, 20} with sequence lengths of 10 kb and 20 kb. Our 15 considered background recombination rates were chosen equidistantly within [0.001, 0.03]. We considered hotspot intensities of {5, 10, 15, 20, 40}-fold the background recombination rate. The length of the hotspots varied among $\{\frac{1}{5}, \frac{1}{10}, \frac{1}{20}, \frac{1}{20}, \frac{1}{35}, \frac{1}{50}\}$-times the sequence length. Due to the large number of resulting setups and the computation times of *LDhelmet* and *LDhat(2)*, we have restricted this analyses to 2 replicates per sample yielding in total 4500 scenarios. We took the RMSE (root mean squared error) as one measure of quality, and approximated it by taking an equidistant grid of 1000 positions along the sequences.
An important tuning parameter of *LDJump* is the number of segments k on which our summary statistics are computed. We chose k between 10 and 50 (yielding segment lengths between 200 and 2000 base pairs depending on the overall sequence length). Figure 2 shows the RMSE depending on the segment length for three different sample sizes. It suggests to choose segments of at least 333 bp. This observation is consistent across the considered sample sizes. The figure also suggests that the performance improves only slightly with larger sample sizes. Our considered type-I error probabilities (between 0.01 and 0.1) did not affect these results.
Table 2 contains an overall comparison in terms of mean, median, and standard deviation of the RMSE for *LDhat* (column 3), *LDhat2* (c. 4), *LDhelmet* (c. 5), and *LDJump* (c. 6-10, with

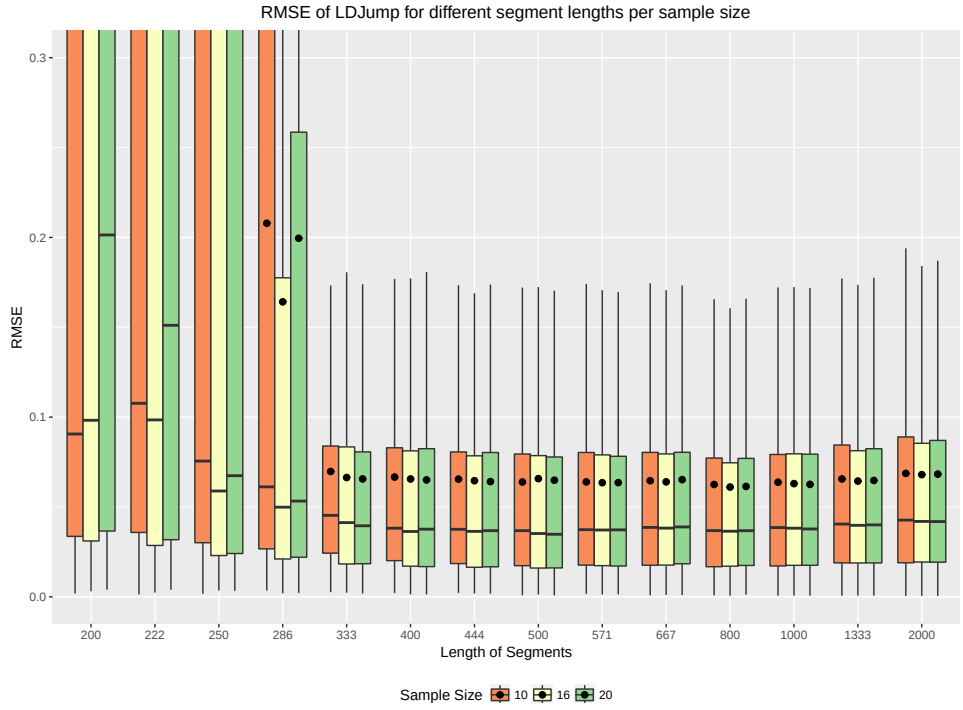

Figure 2: Comparison of the quality of fit of *LDJump* for different segment lengths, distinguished between the considered sample sizes.

different numbers of user-defined segments). The results using different block penalties for *LDhat(2), LDhelmet* along with different type I error probabilities for *LDJump* as 0.1, 0.05, and 0.01 are listed in separate rows.
When choosing a proper number of segments, our method performs equivalently or slightly better than *LDhat2*, and outperforms *LDhat* and also *LDhelmet*. The choice of α did not have a large effect when considering our simple scenarios. Similarly, the block penalty does not much affect the performance of *LDhat2*. This is in contrast to *LDhat* and *LDhelmet* where the choice of the block penalty strongly influences the performance. Higher variability in precision is present between independent estimates from *LDhat* and *LDhelmet* than for *LDJump* and *LDhat2*. With an appropriate number of segments, the standard deviation with *LDJump* is more than 20 % lower than that of *LDhat*, which in turn has a more than 30 % lower SD than *LDhelmet*. We provide a detailed analyses with respect to background rates, sample size, sequence length, hotspot intensity, and hotspot length in appendix B.

3.2.2 Natural Setups

We simulated populations with 16 individuals and sequence lengths of 1Mb. The setups varied in 13 equidistant background-rates between 0.001 and 0.01. The 15 hotspots were evenly distributed along the sequence and had different intensities of 8 to 40-fold compared to the background rate. Every setup was replicated 20 times. We focus on the comparison between

	$\alpha/bpen$	*LDhat*	*LDhat2*	*LDhel*	*LDJump* (Number of Segments) 10	15	20	25	30
$\bar{x}$	0.1/0	0.158	0.064	0.286	0.068	0.065	0.062	0.064	0.066
	0.05/5	0.132	0.064	0.234	0.068	0.065	0.062	0.064	0.066
	0.01/50	0.078	0.064	0.094	0.068	0.065	0.062	0.064	0.066
$x_{0.5}$	0.1/0	0.138	0.036	0.247	0.042	0.040	0.036	0.037	0.040
	0.05/5	0.100	0.036	0.169	0.042	0.040	0.036	0.037	0.040
	0.01/50	0.049	0.036	0.044	0.042	0.040	0.036	0.037	0.040
SD	0.1/0	0.115	0.076	0.227	0.077	0.074	0.096	0.078	0.079
	0.05/5	0.121	0.076	0.224	0.077	0.074	0.096	0.078	0.079
	0.01/50	0.102	0.076	0.145	0.077	0.075	0.096	0.078	0.079

Table 2: Mean ($\bar{x}$), median ($x_{0.5}$) and SD of RMSE for *LDhat*, *LDhat2*, *LDhelmet* (LDhel), and *LDJump* of *simple* setups. Different block penalties (*bpen*), number of segments, and type I error probabilities α are chosen.

LDJump and *LDhat2*, as these methods performed best with the simple scenarios.
Figure 3 provides a comparison of *LDJump* with (grey) and without (purple) bias correction with *LDhat2* (blue). Here, three samples with different background recombination rates of 0.001 (left), 0.0054 (middle), and 0.01 (right) are presented in dotted black lines. Segment lengths were chosen to be 1kb with a quantile in the bias correction of 0.35 (see appendix A.3) and a type-I error probability of 0.05. The bias-correction decreases the bias in the background rates and increases the intensities of the estimated hotspots.

Quality Assessment We looked at the weighted RMSE, defined as WRMSE = $\sqrt{\sum_{i=1}^{n} w_i(\hat{\rho}_i - \rho_i)^2}$, with w_i denoting the length of the estimated segment i divided by the total sequence length. We also consider the proportion of correctly identified hotspots (PCH). A hotspot is counted as correctly identified if it has a non-empty intersection with a detected hotspot (i.e. a region with at least five-fold background recombination rate). The proportion of correctly identified background rates (PCB) has been defined analogously. Finally, the proportion of correctly identified segments is given as PCI = PCH + PCB - 1. PCH and PCB reflect true positives and true negatives, respectively.
We apply *LDJump* with $k = 500$, 1000, 1500, and 2000 segments and estimate the recombination maps using quantiles of 0.25, 0.35, 0.45, and 0.5 in the bias correction (see appendix A.3)). Hence, we can identify the best combination of bias correction and segment lengths. Notice that the short segments are 2kb, 1kb, 666 and 500 bp long and hotspot lengths vary in the setup between 1 and 2kb. Therefore, the scenario with k =1500 is more challenging as the hotspot boundaries will systematically differ from the segment boundaries. A direct comparison with *LDhat2* using a block penalty of 50 (based on the results from the *simple setups*) is provided. The different choices of k are displayed by the first four groups of boxplots in Figure 4. For each of these four groups, quantiles of 0.25, 0.35, 0.4, and 0.5 are used in the bias correction and

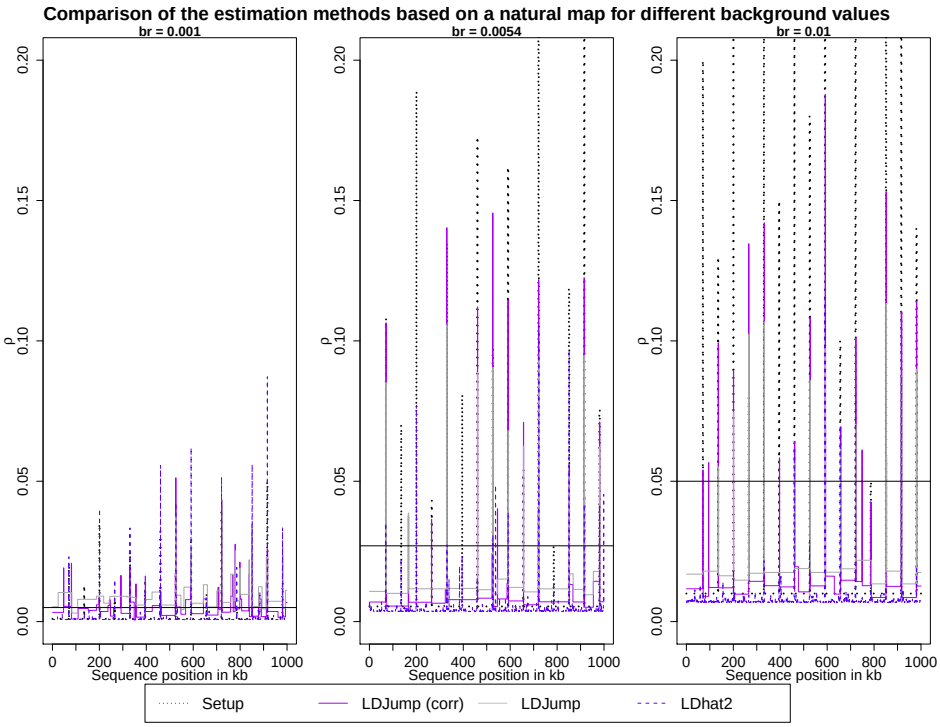

Figure 3: Estimated recombination maps from *LDJump*, improvements with the simulation based bias-correction and *LDhat2* for setups differing in the background rates of 0.001 (left), 0.0054 (middle), and 0.01 (right). Setups (black dotted lines) simulated for these comparisons contain 16 hotspots. Horizontal lines represent the hotspot threshold (5·background rate).

are presented in different colors. The rightmost bar per panel (in blue) summarizes the result of *LDhat2*. From top-left to bottom-right, we show PCH, PCB, PCI, the estimated number of blocks and the weighted RMSE.

PCH may be interpreted as a measure of sensitivity, whereas PCB provides a measure of specificity. We can see that our method has very high detection rates across k with even less variability in performance than *LDhat2*. On the other hand, *LDhat2* has very high PCB proportions. The best PCB values for *LDJump* are obtained for the smallest quantile.

As an overall measure, we display the sum of PCH and PCB minus one as PCI in the bottom-left panel. It turns out that PCI is larger for *LDJump* regardless of the tuning parameters. In the bottom-middle panel we can see that the number of estimated block of *LDJump* depends on k. When using 500 segments, the estimated number of blocks is still below 31, the true number of blocks in the recombination map (due to 15 hotspots). For larger k the number of blocks is slightly overestimated. *LDhat2* estimated many more blocks, indeed the number of change points in recombination tended to be larger by a factor of more than 3000. The bottom-right plot shows the weighted RMSE as an overall quality measure showing a similar level of accuracy across k and compared with *LDhat2*. A more detailed investigation reveals that our method estimates hotspot rates more precisely, but provides less accurate estimators of the background recombination rate.

Our results also show that our method is fairly robust with respect to tuning choices. This is also true for $k = 1500$, where there the hotspots have an unfavorable location. To obtain a reasonable tradeoff between sensitivity (PCH) and specificity (PCB), segment lengths of 1kb (based on 1000 segments of sequence length 1Mb) and a quantile of 0.35 in the bias correction seem to be a good choice with *LDJump*.

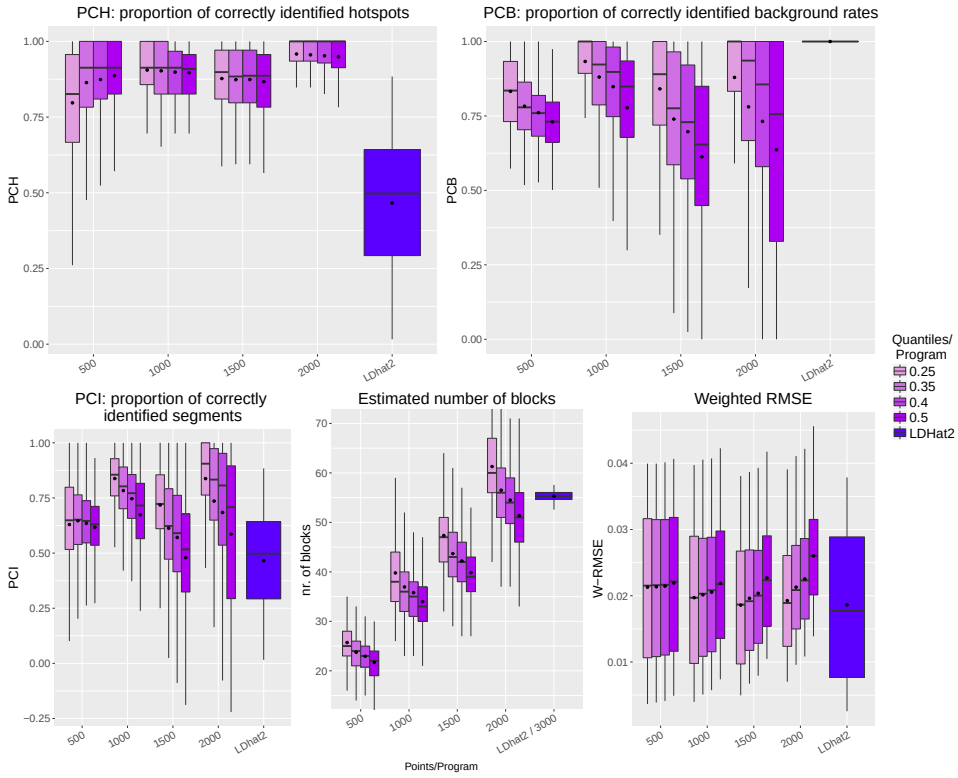

Figure 4: Quality assessment is performed based on the proportion of correctly identified hotspots (PCH, top-left), the proportion of correctly identified background rates (PCB, top-right), the proportion of correctly identified segments (PCI = PCH+PCB-1, bottom-left), the estimated number of blocks (bottom-middle), and the weighted RMSE (bottom-right). Based on 13 setups with 20 replicates these measures are computed for *LDJump* using different number of initial segments k (500, 1000, 1500, 2000) and compared with the results of *LDhat2* using a block penalty of 50.

Figure 5 shows our considered quality measures depending on the background recombination rates. We provide the average performance of 20 replicates. We can see that *LDhat2* has constant PCB and decreasing PCH as the background rate increases. *LDJump* shows constant values for PCH and slightly increasing PCB for higher background rates. The overall measure PCI behaves as expected given these observations (PCI and PCH overlap for *LDhat2*). Ten times the weighted RMSE is also plotted. It can be seen that *LDhat2* leads to slightly smaller weighted RMSE values with decreasing differences for larger ρ.

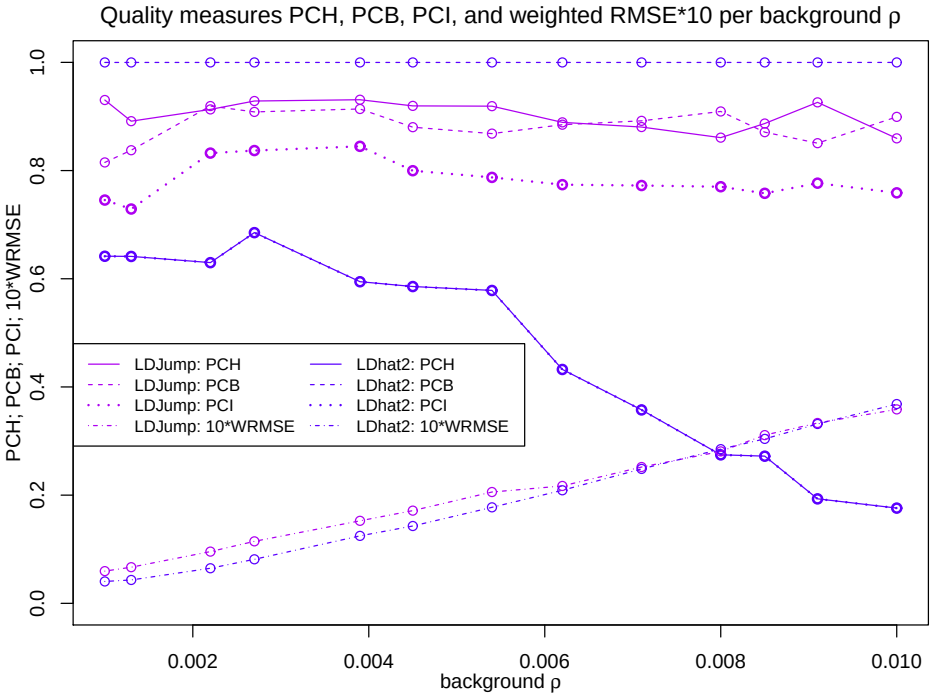

Figure 5: Proportion of correctly identified hotspots (PCH, solid), proportion of correctly identified background rates (PCB, dashed), the sum of these two quality measures-1 (PCI, dotted), and weighted RMSE*10 (dash-dotted) across different recombination rates. We compare *LDJump* (purple, segment length: 1kb, quantile 0.35), with *LDhat2* (blue, same line coding per quality measure).

3.3 Runtime

Runtime is an important aspect of any method, especially for larger numbers of sequences, and separate analyses for several populations. Hence, we provide a comparison with respect to runtime (in seconds) between *LDhat*, *LDhat2*, *LDhelmet*, and *LDJump*. Again, we looked at different block penalty choices, as well as at different numbers of atomic segments k for *LDJump* in Table 3. As summaries, we computed the mean (top), median (middle), and SD (bottom) of our measured runtimes. We can see that especially *LDhat2* and *LDhelmet* require ten to forty times longer runtimes than *LDJump*. While being only slightly slower, we have seen before that *LDhat* leads to considerably less accurate estimates.

In Table 4 we explore the effects of sample size and sequence length on the runtime. We compared the aforementioned methods with respect to their mean and median runtimes again for our *simple* setups. The runtimes for *LDhat2* and *LDhelmet* are strongly affected by sequence length and sample size. Interestingly *LDhat2* seems to have more problems dealing with longer sequences, whereas *LDhelmet* shows an especially large increase in runtime when the sample size increases. The runtime of *LDJump* (using segments of length 500 and 1000 bp) seems less sensitive to such increases, but more sensitive than *LDhat* w.r.t. sequence length.

	LDhat			*LDhat2*			*LDhelmet*			*LDJump*			
	0	5	50	0	5	50	0	5	50	10	15	20	25
$\bar{x}$	33	53	138	649	2366	2324	1271	1345	1769	62	47	41	39
$x_{0.5}$	33	52	123	619	1934	1909	814	880	1287	51	40	36	34
SD	6	8	61	256	1228	1195	1063	1070	1144	37	26	20	16

Table 3: Mean ($\bar{x}$), median ($x_{0.5}$), and SD of runtime (in seconds) for *LDhat*, *LDhat2*, *LDhelmet*, and *LDJump* under simple setups. For each method, separate columns provide values depending on either the block penalty (columns 2-10), or the number of predefined segments on which *LDJump* was applied (columns 11-15).

Time	Method	Sample Size			Sequence Length	
		10	16/10	20/10	10kb	20kb/10kb
Mean	*LDhat*	124	14%	20%	121.42	28%
Mean	*LDhat2*	1862	33%	41%	1387.72	135%
Mean	*LDhelmet*	709	90%	358%	1581.01	24%
Mean	*LDJump*	34	16%	25%	28.58	67%
Median	*LDhat*	109	15%	24%	111.41	24%
Median	*LDhat2*	1526	29%	36%	1398.45	133%
Median	*LDhelmet*	625	95%	416%	1101.45	43%
Median	*LDJump*	31	19%	21%	28.41	68%

Table 4: Runtime in seconds and ratios between sample sizes and sequence lengths are provided. We computed the mean and median runtime for each method and scenario. The effect of increasing sample sizes or sequence lengths is shown in percent.

In Table 5 we show the mean, median, and SD of runtimes in seconds based on *natural* setups. On average *LDJump* turns out to be about ten to twenty times faster than *LDhat2*. Choosing larger values of k reduces the runtime for *LDJump*. Due to the faster computation of certain summary statistics, runtime was reduced by about 50% when going from 500 to 2000 segments. The remarkable difference between median and mean for *LDhat2*, is caused by different recombination rates. In appendix C we compare the development of runtime across background rates and different k and see that our method is approximately constant with respect to ρ in contrast to *LDhat2*. Overall, *LDJump* provides a particularly attractive combination of performance and runtime.

4 Application

We sampled the same 103kb region between SNPs rs10622653 and rs2299784 as characterized by [Tiemann-Boege et al., 2006] with sperm typing containing the PCP4 gene. Specifically, we used 50 individuals of 4 subpopulations from 4 European regions (TSI, FIN, IBS, GBR) with

	LDhat2	*LDJump*			
		500	1000	1500	2000
$\bar{x}$	77237	6758	4281	3575	3463
$x_{0.5}$	122396	6809	4327	3592	3471
SD	2434	528	336	298	308

Table 5: Mean ($\bar{x}$), median ($x_{0.5}$), and SD of runtime (in seconds) for *LDhat2* and *LDJump* for natural setups. For *LDJump* we provide values depending on the number of predefined segments.

data taken from [The 1000 Genomes Project Consortium, 2015]. The data has been reformatted from `vcf`-format to `fasta`-files with the `R` packages [Knaus and Grünwald, 2017, Paradis et al., 2004] using two sequences per (diploid) sample and the reference sequence 80.37 (GRCH37) from [The 1000 Genomes Project Consortium, 2015]. We applied *LDJump* with a segment length of 1kb and chose the 35%-quantile for the bias-correction. The estimated recombination map for the Italian population (TSI, using a lookup table of 100 sequences and a mutation rate of 0.005) in the top-left panel of Figure 6 is similar to the measures obtained by experimental data (sperm typing) in [Tiemann-Boege et al., 2006] (see bottom-left panel of Figure 6) in the region from 60-100kb. We also compared this region with the double strand break maps (representing active recombination hotspots) from [Pratto et al., 2014], see Figure 6 bottom-right. The computationally inferred, historical hotspots estimated with LDJump in this region (60-100kb) also agree with the DSB activity measured by [Pratto et al., 2014]. However, we additionally estimated with *LDJump* a hotspot before the PCP4 gene around position 45kb. This hotspot was also found by other LD-based algorithms [McVean et al., 2004, Li and Stephens, 2003], see Figure 6, bottom-left [Tiemann-Boege et al., 2006].
Given the lack of active recombination in this region (absence of this hotspot in the DSB maps for the 2 European donors (carrying the PRDM9 allele A) or the donor with African descend (carrying the PRDM9 allele C)), we hypothesize that the observed hotspot using *LDJump* at position ∼45kb might represent a historical hotspot that got extinct. Alternatively, it could be a population-specific hotspot given that its intensity varies among different European populations. In order to test this latter hypothesis, active recombination maps from different populations would be needed. However, one can also see differences in DSB intensities between individuals of the same populations (e.g. hotspot at position 95kb present only in the individual with a PRDM9 C allele) suggesting that the intensity of hotspots is highly variable.
Differences between hotspot rates estimated from LD patterns compared to estimates based on sperm typing have also been observed by [Jeffreys and Neumann, 2009]. This might be caused by the short life-span of hotspots and their rapid evolution in intensity and genomic position among populations and species [Coop and Myers, 2007, Myers et al., 2010, Jeffreys et al., 2013]. Fine-scale population specific differences with respect to recombination events have been highlighted in studies such as [Kong et al., 2010, Berg et al., 2011, Fledel-Alon et al., 2011, Pratto et al., 2014]. In the top-right panel of Figure 6 we provide an estimated recombination map using *LDJump* on the Hap Map data samples mentioned above. Also for

historical recombination events, we see population-specific differences in the detected hotspots. Interestingly, all considered populations have detected the aforenamed hotspot before the PCP-4 region (∼45kb), which was not found by sperm-typing or DSB-mapping.

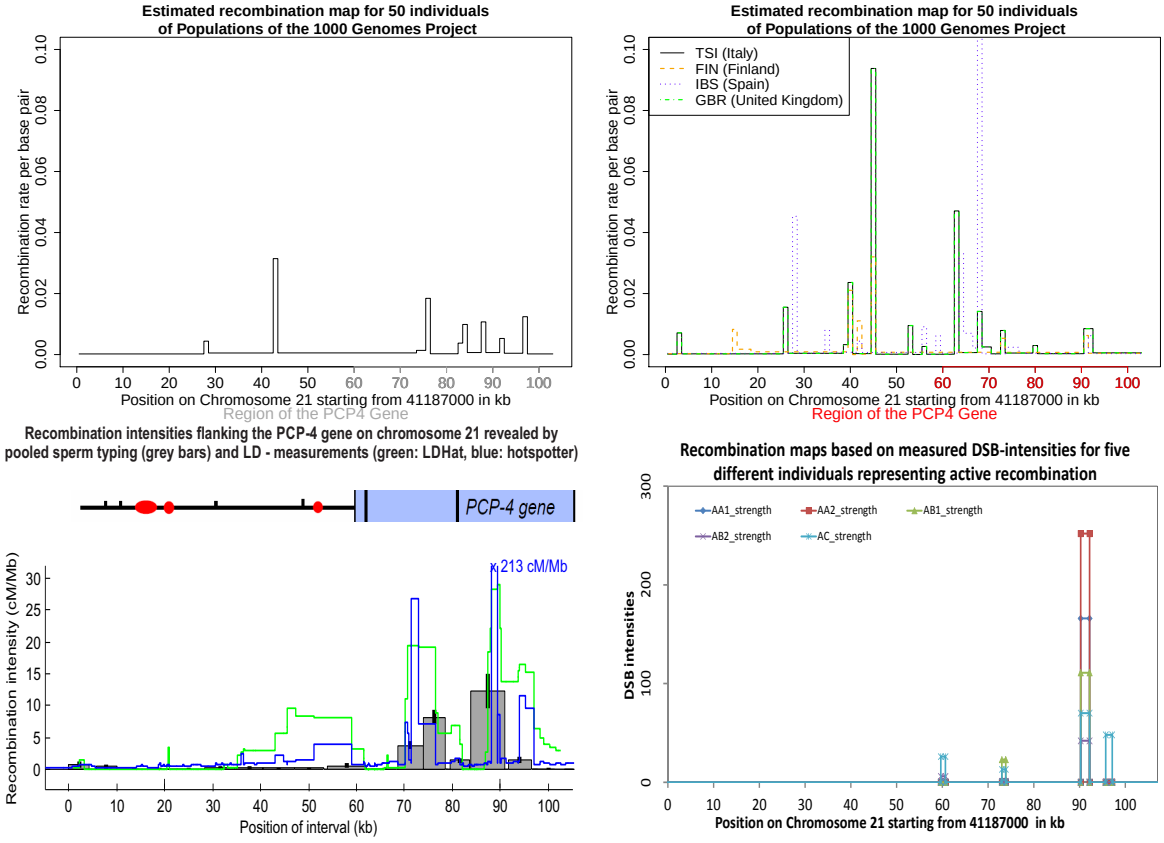

Figure 6: Top-left: Estimated recombination map of 50 individuals from an Italian population on Chromosome 21:41187000-41290679, including the PCP4-Gene (red region on the x-axis). Top-right: Estimated recombination map of 4 different European populations (Italy, Finland, Spain, and United Kingdom) on Chromosome 21:41187000-41290679, including the PCP4-Gene (in red). Bottom-left: Estimated recombination map of the same 103kb region including PCP4 on Chromosome 21 taken from [Tiemann-Boege et al., 2006] based on sperm typing 13 intervals ∼ 5kb in size (grey boxes), and LD-inferred measures (green lines: *LDhat* and blue line: Hotspotter). Bottom-right: Recombination maps based on measured double strand break (DSB) intensities for five different individuals representing active recombination from [Pratto et al., 2014].

5 Conclusion

We introduce a new method called *LDJump* to estimate the heterogeneity of recombination rates across the DNA sequence from population genetic data. A sequence is divided into segments of proper length in a first step. Subsequently, we use a generalized additive regression model to estimate the constant recombination rates per segment. Then, we apply a simultaneous multiscale change-point estimator (SMUCE) to estimate the breakpoints in the recombination rates across the sequence. We provide detailed comparisons of our method with the recent reversible jump MCMC methods *LDhat(2)* and *LDhelmet*. Our estimates are very fast, perform favourably in the detection of hotspots, and show similar accuracy levels as the best available competitor for *simple* and *natural* setups, respectively. We have applied our method on several human populations (data taken from the 1000 Genomes project) and compared the estimated hotspots with recombination intensities measured by sperm-typing data and double strand break maps. These computations revealed population specific hotspots in the region surrounding the PCP4-gene located on Chromosome 21. We have implemented our approach in the `R`-package *LDJump*, which can be downloaded from https://github.com/PhHermann/LDJump. We recommend users to apply our method with segment lengths of 1kb and a bias correction using the default quantile of 0.35.

Acknowledgements

We are grateful to Kerstin Gärtner and Renato Pereira Salazar for their assistance with the software packages *LDhat* and *LDhelmet* as well as the data acquisition for the application, respectively. We are thankful to Katharina Sallinger, Renato Pereira Salazar, and Theresa Schwarz for their helpful comments. This work was supported by the 'Austrian Science Fund' (FWF) P27698-B22 to I.T-B., and the DOC Fellowship of the Austrian Academy of Sciences (24529) of A.H.

References

[Arnheim et al., 2007] Arnheim, N., Calabrese, P., and Tiemann-Boege, I. (2007). Mammalian Meiotic Recombination Hot Spots. Annual Review of Genetics, 41(1):369–399.

[Auton et al., 2012] Auton, A., Fledel-Alon, A., Pfeifer, S., Venn, O., Segurel, L., Street, T., Leffler, E. M., Bowden, R., Aneas, I., Broxholme, J., Humburg, P., Iqbal, Z., Lunter, G., Maller, J., Hernandez, R. D., Melton, C., Venkat, A., Nobrega, M. A., Bontrop, R., Myers, S., Donnelly, P., Przeworski, M., and McVean, G. (2012). A Fine-Scale Chimpanzee Genetic Map from Population Sequencing. Science, 336(6078):193–198.

[Auton and McVean, 2007] Auton, A. and McVean, G. (2007). Recombination rate estimation in the presence of hotspots. Genome Research, 17(8):1219–1227.

[Batorsky et al., 2011] Batorsky, R., Kearney, M. F., Palmer, S. E., Maldarelli, F., Rouzine, I. M., and Coffin, J. M. (2011). Estimate of effective recombination rate and average selection coefficient for HIV in chronic infection. Proceedings of the National Academy of Sciences of the United States of America, 108(14):5661–6.

[Baudat et al., 2013] Baudat, F., Imai, Y., and de Massy, B. (2013). Meiotic recombination in mammals: localization and regulation. Nature reviews. Genetics, 14(11):794–806.

[Berg et al., 2010] Berg, I. L., Neumann, R., Lam, K.-W. G., Sarbajna, S., Odenthal-Hesse, L., May, C. A., and Jeffreys, A. J. (2010). PRDM9 variation strongly influences recombination hot-spot activity and meiotic instability in humans. Nature Genetics, 42(10):859–863.

[Berg et al., 2011] Berg, I. L., Neumann, R., Sarbajna, S., Odenthal-Hesse, L., Butler, N. J., and Jeffreys, A. J. (2011). Variants of the protein PRDM9 differentially regulate a set of human meiotic recombination hotspots highly active in African populations. Proc Natl Acad Sci U S A, 108(30):12378–12383.

[Birdsell, 2002] Birdsell, J. A. (2002). Integrating Genomics, Bioinformatics, and Classical Genetics to Study the Effects of Recombination on Genome Evolution. Molecular Biology and Evolution, 19(7):1181–1197.

[Box and Cox, 1964] Box, G. E. P. and Cox, D. R. (1964). An analysis of transformations. Journal of the Royal Statistical Society. Series B (Methodological, pages 211–252.

[Chan et al., 2012] Chan, A. H., Jenkins, P. A., and Song, Y. S. (2012). Genome-Wide Fine-Scale Recombination Rate Variation in Drosophila melanogaster. PLoS Genetics, 8(12):e1003090.

[Coop and Myers, 2007] Coop, G. and Myers, S. R. (2007). Live Hot, Die Young: Transmission Distortion in Recombination Hotspots. PLoS Genetics, 3(3):e35.

[Coop et al., 2008] Coop, G., Wen, X., Ober, C., Pritchard, J. K., and Przeworski, M. (2008). High-Resolution Mapping of Crossovers Reveals Extensive Variation in Fine-Scale Recombination Patterns Among Humans. Science, 319(5868):1395–1398.

[Duret and Galtier, 2009] Duret, L. and Galtier, N. (2009). Biased Gene Conversion and the Evolution of Mammalian Genomic Landscapes. Annual Review of Genomics and Human Genetics, 10(1):285–311.

[Fearnhead and Donnelly, 2001] Fearnhead, P. and Donnelly, P. (2001). Estimating recombination rates from population genetic data. Genetics, 159:1299–1318.

[Fearnhead and Donnelly, 2002] Fearnhead, P. and Donnelly, P. (2002). Approximate Likelihood Methods for Estimating Local Recombination Rates. Journal of the Royal Statistical Society. Series B (Statistical Methodology), 64(4):657–680.

[Fledel-Alon et al., 2011] Fledel-Alon, A., Leffler, E. M., Guan, Y., Stephens, M., Coop, G., and Przeworski, M. (2011). Variation in human recombination rates and its genetic determinants. PLoS ONE, 6(6).

[Frick et al., 2014] Frick, K., Munk, A., and Sieling, H. (2014). Multiscale change-point inference. Journal of the Royal Statistical Society: Series B, 76(3):495–580.

[Futschik et al., 2014] Futschik, A., Hotz, T., Munk, A., and Sieling, H. (2014). Multiscale DNA partitioning: Statistical evidence for segments. Bioinformatics, 30(16):2255–2262.

[Gao et al., 2016] Gao, F., Ming, C., Hu, W., and Li, H. (2016). New Software for the Fast Estimation of Population Recombination Rates (FastEPRR) in the Genomic Era. G3 (Bethesda, Md.), 6(6):1563–1571.

[Gärtner and Futschik, 2016] Gärtner, K. and Futschik, A. (2016). Improved Versions of Common Estimators of the Recombination Rate. Journal of Computational Biology, 23(9):756–768.

[Halldorsson et al., 2016] Halldorsson, B. V., Hardarson, M. T., Kehr, B., Styrkarsdottir, U., Gylfason, A., Thorleifsson, G., Zink, F., Jonasdottir, A., Jonasdottir, A., Sulem, P., Masson, G., Thorsteinsdottir, U., Helgason, A., Kong, A., Gudbjartsson, D. F., and Stefansson, K. (2016). The rate of meiotic gene conversion varies by sex and age. Nat Genet, 48(11):1377–1384.

[Haubold and Pfaffelhuber, 2013] Haubold, B. and Pfaffelhuber, P. (2013). ms2dna, v. 1.16: Convert Simulated Haplotype Data to DNA Sequences.

[Hill and Robertson, 1968] Hill, W. G. and Robertson, A. (1968). Linkage disequilibrium in finite populations. Theoretical and Applied Genetics, 38(6):226–231.

[Hotz and Sieling, 2016] Hotz, T. and Sieling, H. (2016). stepR: Fitting Step-Functions.

[Hudson, 1987] Hudson, R. R. (1987). Estimating the recombination parameter of a finite population model without selection. Genetical research, 50(2007):245–250.

[Hudson, 2001] Hudson, R. R. (2001). Two-locus sampling distributions and their application. Genetics, 159(4):1805–1817.

[Hudson and Kaplan, 1985] Hudson, R. R. and Kaplan, N. L. (1985). Statistical properties of the number of recombination events in the history of a sample of DNA sequences. Genetics, 111(1):147–164.

[Jeffreys et al., 2013] Jeffreys, A. J., Cotton, V. E., Neumann, R., and Lam, K.-W. G. (2013). Recombination regulator PRDM9 influences the instability of its own coding sequence in humans. Proceedings of the National Academy of Sciences, 110(2):600–605.

[Jeffreys et al., 2001] Jeffreys, A. J., Kauppi, L., and Neumann, R. (2001). No Title. Nature Genetics, 29(2):217–222.

[Jeffreys and Neumann, 2009] Jeffreys, A. J. and Neumann, R. (2009). The rise and fall of a human recombination hot spot. Nature genetics, 41(5):625–9.

[Jombart, 2008] Jombart, T. (2008). adegenet: a R package for the multivariate analysis of genetic markers. Bioinformatics, 24(11):1403–1405.

[Kamm et al., 2016] Kamm, J. A., Spence, J. P., Chan, J., and Song, Y. S. (2016). Two-Locus Likelihoods Under Variable Population Size and Fine-Scale Recombination Rate Estimation. Genetics, 203(3):1381 LP – 1399.

[Knaus and Grünwald, 2017] Knaus, B. J. and Grünwald, N. J. (2017). vcfr : a package to manipulate and visualize variant call format data in R. Molecular Ecology Resources, 17(1):44–53.

[Kong et al., 2010] Kong, A., Thorleifsson, G., Gudbjartsson, D. F., Masson, G., Sigurdsson, A., Jonasdottir, A., Walters, G. B., Jonasdottir, A., Gylfason, A., Kristinsson, K. T., Gudjonsson, S. A., Frigge, M. L., Helgason, A., Thorsteinsdottir, U., and Stefansson, K. (2010). Fine-scale recombination rate differences between sexes, populations and individuals. Nature, 467(7319):1099–1103.

[Kuhner et al., 2000] Kuhner, M. K., Yamato, J., and Felsenstein, J. (2000). Maximum likelihood estimation of recombination rates from population data. Genetics, 156(3):1393–1401.

[Li and Stephens, 2003] Li, N. and Stephens, M. (2003). Modeling Linkage Disequilibrium and Identifying Recombination Hotspots Using Single-Nucleotide Polymorphism Data. Genetics, 165(4):2213–2233.

[Lin et al., 2013] Lin, K., Futschik, A., and Li, H. (2013). A Fast Estimate for the Population Recombination Rate Based on Regression. Genetics, 194(2):473–484.

[McVean et al., 2002] McVean, G. A. T., Awadalla, P., and Fearnhead, P. (2002). A coalescent-based method for detecting and estimating recombination from gene sequences. Genetics, 160(3):1231–41.

[McVean et al., 2004] McVean, G. A. T., Myers, S. R., Hunt, S., Deloukas, P., Bentley, D. R., and Donnelly, P. (2004). The fine-scale structure of recombination rate variation in the human genome. Science, 304(5670):581–584.

[Myers et al., 2005] Myers, S., Bottolo, L., Freeman, C., McVean, G., and Donnelly, P. (2005). A Fine-Scale Map of Recombination Rates and Hotspots Across the Human Genome. Science, 310(5746):321–324.

[Myers et al., 2010] Myers, S., Bowden, R., Tumian, A., Bontrop, R. E., Freeman, C., MacFie, T. S., McVean, G., and Donnelly, P. (2010). Drive Against Hotspot Motifs in Primates Implicates the PRDM9 Gene in Meiotic Recombination. Science, 327(5967):876–879.

[Myers and Griffiths, 2003] Myers, S. R. and Griffiths, R. C. (2003). Bounds on the minimum number of recombination events in a sample history. Genetics, 163(1):375–394.

[Niu et al., 2016] Niu, Y. S., Hao, N., and Zhang, H. (2016). Multiple Change-Point Detection: A Selective Overview. Statistical Science, 31(4):611–623.

[Pages et al., 2016] Pages, H., Aboyoun, P., Gentleman, R., and DebRoy, S. (2016). Biostrings: String objects representing biological sequences, and matching algorithms.

[Paradis et al., 2004] Paradis, E., Claude, J., and Strimmer, K. (2004). A{PE}: analyses of phylogenetics and evolution in {R} language. Bioinformatics, 20:289–290.

[Pratto et al., 2014] Pratto, F., Brick, K., Khil, P., Smagulova, F., Petukhova, G. V., and Camerini-Otero, R. D. (2014). Recombination initiation maps of individual human genomes. Science, 346(6211).

[R Core Team, 2017] R Core Team (2017). R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria.

[Reid, 2013] Reid, N. (2013). Aspects of likelihood inference. Bernoulli, 19(4):1404–1418.

[Sabeti et al., 2006] Sabeti, P. C., Schaffner, S. F., Fry, B., Lohmueller, J., Varilly, P., Shamovsky, O., Palma, A., Mikkelsen, T. S., Altshuler, D., and Lander, E. S. (2006). Positive natural selection in the human lineage. Science (New York, N.Y.), 312(5780):1614–20.

[Shannon, 1948] Shannon, C. E. (1948). A Mathematical Theory of Communication. Bell System Technical Journal, 5(3):3.

[Smagulova et al., 2011] Smagulova, F., Gregoretti, I. V., Brick, K., Khil, P., Camerini-Otero, R. D., and Petukhova, G. V. (2011). Genome-wide analysis reveals novel molecular features of mouse recombination hotspots. Nature, 472(7343):375–378.

[Staab et al., 2014] Staab, P. R., Zhu, S., Metzler, D., and Lunter, G. (2014). Scrm: Efficiently simulating long sequences using the approximated coalescent with recombination. Bioinformatics, 31(10):1680–1682.

[The 1000 Genomes Project Consortium, 2015] The 1000 Genomes Project Consortium (2015). A global reference for human genetic variation. Nature, 526(7571):68–74.

[Tiemann-Boege et al., 2006] Tiemann-Boege, I., Calabrese, P., Cochran, D. M., Sokol, R., and Arnheim, N. (2006). High-Resolution Recombination Patterns in a Region of Human Chromosome 21 Measured by Sperm Typing. PLoS Genetics, 2(5):e70.

[Varin et al., 2011] Varin, C., Reid, N., and Firth, D. (2011). An Overview of Composite Likelihood Methods. Statistica Sinica, 21(1):5–42.

[Wall, 2000] Wall, J. D. (2000). A Comparison of Estimators of the Population Recombination Rate. Mol. Biol. Evol, 17(1):156–163.

[Wall, 2004] Wall, J. D. (2004). Estimating recombination rates using three-site likelihoods. Genetics, 167(3):1461–1473.

[Warnes et al., 2013] Warnes, G., Gorjanc, G., Leisch, F., and Man, M. (2013). genetics: Population Genetics.

[Wiuf, 2002] Wiuf, C. (2002). On the minimum number of topologies explaining a sample of DNA sequences. Theoretical Population Biology, 62(4):357–363.

[Wood, 2011] Wood, S. N. (2011). Fast stable restricted maximum likelihood and marginal likelihood estimation of semiparametric generalized linear models. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 73(1):3–36.

A Regression Model

Data management for computing the summary statistics was conducted with the functions `readDNAStringSet`, `writeXStringSet` of the R-package [Pages et al., 2016] and the functions `genind2df`, `DNAbin2genind` of the R-package `adegenet` [Jombart, 2008].
We simulated populations with {10, 16, 20} individuals and sequence lengths of {500, 1000, 2000, 3000, 5000} bp. The recombination rates per base pair were simulated from a uniform distribution in the intervals {[0,0.01], [0.01,0.02], [0.02,0.05], [0.05,0.1], [0.1,0.2]} and used for every combination of population size and sequence length. For the first setup we simulate 100 recombination rates between [0,0.01]. Subsequently, a population with 10 individuals and a sequence length of 500 nucleotides is simulated using every of these 100 values. This procedure is conducted with all setups combining population sizes and sequence lengths. Hence, the simulated data set consists of 8000 observations, where 1 observation was removed due to the lack of a recombination and mutation event. The mutation rate was set 0.01 per base pair for all simulations as well as computations with *LDhat* and *LDhelmet*. We compute the summary statistics for every observation in our data set and regressed the summary statistics on the known recombination rates. We also took linear and quadratic effects of summary statistics into account and performed variable selection based on ANOVA. No significant effect was estimated for GC content (*gcco*), although in natural populations GC-biased gene conversion is associated with recombination [Birdsell, 2002] and modifies the GC content in regions with active recombination (reviewed in [Duret and Galtier, 2009]). Our analysis of the simulated GC content per recombination rates showed an equal distribution of the GC content among the simulated range of ρ.

A.1 Coefficients & Effect Plots

Figure 7 contains graphical representations of the effects of the summary statistics. The first six plots from top-left to bottom-right represent the estimated cubic spline functions for the variables *vapw, apwd, hahe, rsqu, ldpr, wath*, and the last two plots show the estimated quadratic effects of *hats* and *fgts*. The 95% confidence interval of the effect is plotted with dashed lines. Table 6 contains the estimation results of the regression model for the summary statistics with significant effects (represented with asterisks). The first two columns show the coefficients and the standard deviation of the quadratic functions and columns three and four the *EDF* and *ref.df* of the cubic spline functions. The quality of fit measure R^2 (0.74) shows a high model fit based on the simulated data.

A.2 On the model assumptions of variance homogeneity and normality

Here, we denote the Box-Cox transformation [Box and Cox, 1964] (2) as

$$t(\rho, \gamma, \epsilon) = \begin{cases} \frac{(\rho+\epsilon)^\gamma - 1}{\gamma} & \text{for } \gamma \neq 0 \\ \ln(\rho + \epsilon) & \text{for } \gamma = 0. \end{cases} \tag{2}$$

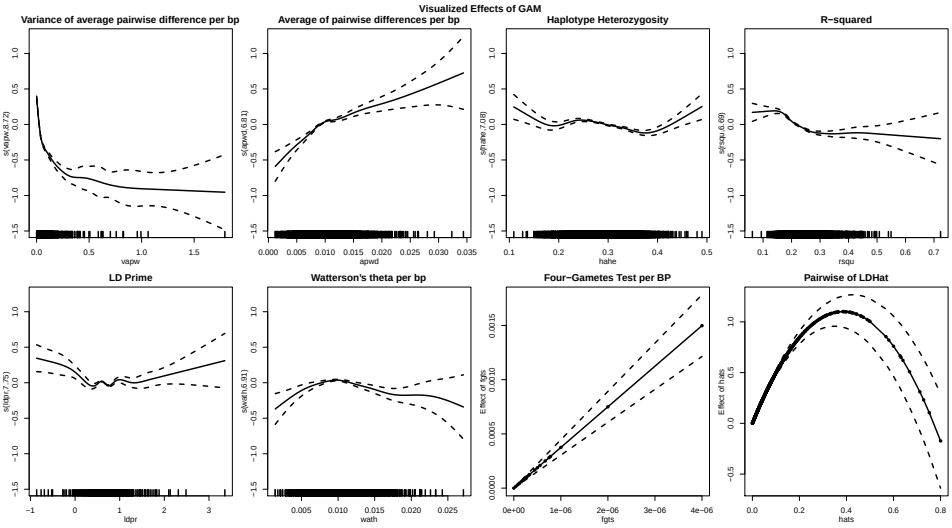

Figure 7: Visualized effects of the significant variables estimating the recombination rates via a generalized additive model.

This transformation performed best under the considered transformations such as logarithmic or exponential transformations. In order to tune the model with respect to homogeneity and normality of the residuals as well as high prediction accuracy, we compared the performance of different (combinations of) parameters of this transformation. The considered grid of values for γ and ϵ for the Box-Cox transformation (2) was $\{0.001, 0.01, 0.05, 0.1, 0.25, 0.5, 0.75, 1\}$ and $\{0, 0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07, 0.08, 0.09, 0.10, 0.15\}$, respectively.
In place of this comparison the plots of Figure 8 are produced with the chosen model and the trained data. The left plot of Figure 8 shows the scatter plot of the predicted values (y-axis) and the true values (x-axis) of the simulated recombination rates, both in transformed scale. By dividing the grid of recombination rates into 15 segments we can compute the standard deviations for the predictions in this interval. The ratio of the standard deviation by the mean of the standard deviations of all intervals is visualized in the middle plot of Figure 8 with an estimated smoothing spline. Here, approximately 15% lower and 15% higher standard deviation of the predictions in $[0.0148, 0.0259]$ and $[0.1429, 0.1571]$ are computed compared to the mean of standard deviations over all intervals. The number of segments is arbitrarily chosen and computations with 10 to 25 segments show robustness with a maximum deviation to the mean of 20%. The right plot of Figure 8 shows the QQ-plot for the residuals of the model.
Figure 9 contains heat maps for the model assumption criteria of the GAM models. Each panel has its own color key and is calibrated that greener boxes are better performances in terms of the criterion. The x-axis of each plot contains the values of ϵ and the y-axis the values of γ. The top-left and the top-right panel show the sum of squared and the sum of absolute differences of the standard deviation to their means, respectively. These measures rely on the computations visualized with the smoothing spline in middle plot of Figure 8. Naturally, smaller values indicate a better performance in terms of variance homogeneity and are coded with (darker) green. Greater sums of squared/absolute differences are visualized with brighter colors. Values

linear/ quadratic	Coefficients (SD)	cubic splines	EDF (Ref.df)
(Intercept)	-2.52 (0.01)***	s(vapw)	8.72 (9.00)***
hats	5.71 (0.12)***	s(apwd)	6.81 (9.00)***
hats^2	-7.41 (0.22)***	s(hahe)	7.08 (9.00)***
fgts	375.32 (36.17)***	s(rsqu)	6.69 (9.00)***
fgts^2	-197694.71 (29144.10)***	s(ldpr)	7.75 (9.00)***
		s(wath)	6.91 (9.00)***
	R^2	0.74	
	Num. obs.	7999	

$^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$

Table 6: Coefficients of summary statistics estimated via a generalized additive model to explain the recombination rate.

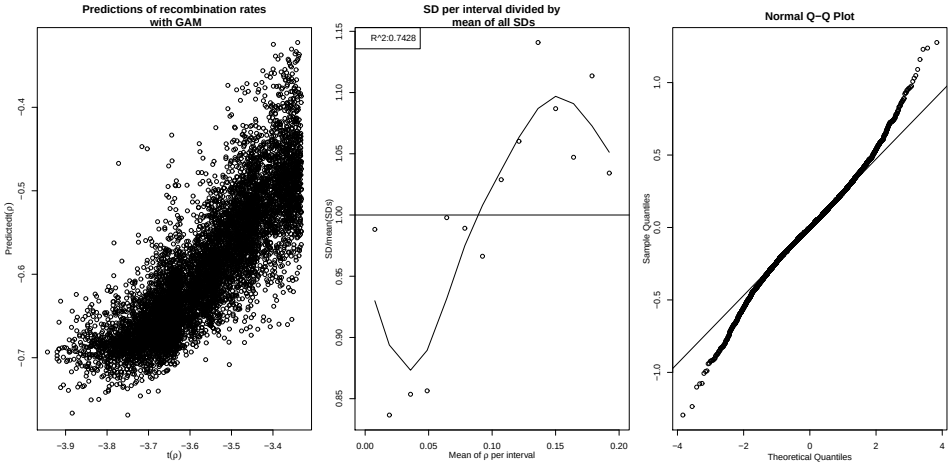

Figure 8: Plot of predicted versus true values (left), graphical tests for variance homogeneity (middle) and normality of residuals (right) of the chosen GAM model

of γ in a range of 0.05 - 0.35 with $\epsilon = 0$ and γ in a range of 0.001 - 0.1 with $\epsilon = 0.1$ are seen as possible candidates for a proper transformation.
Normality of the residuals is considered in the bottom-left panel. Here, Shapiro-Wilk statistics are calculated with values close to 1 coded in green color. The standard implementation of this test in [R Core Team, 2017] is restricted to 5000 observations. Therefore, we drew 100 different samples of 5000 residuals and computed the mean of these 100 Shapiro-Wilk statistics. We can observe a similar pattern as for the variance homogeneity comparisons except for the combination $\gamma = 0.1$ and $\epsilon = 0.1$. The quality of the regression model in terms of R^2 also points to the same choice of the parameters γ and ϵ. We decide for the setup $\gamma = 0.25$ and $\epsilon = 0$ due to the better performance in terms of normality and a higher R^2 given very similar values for the variance homogeneity measures.

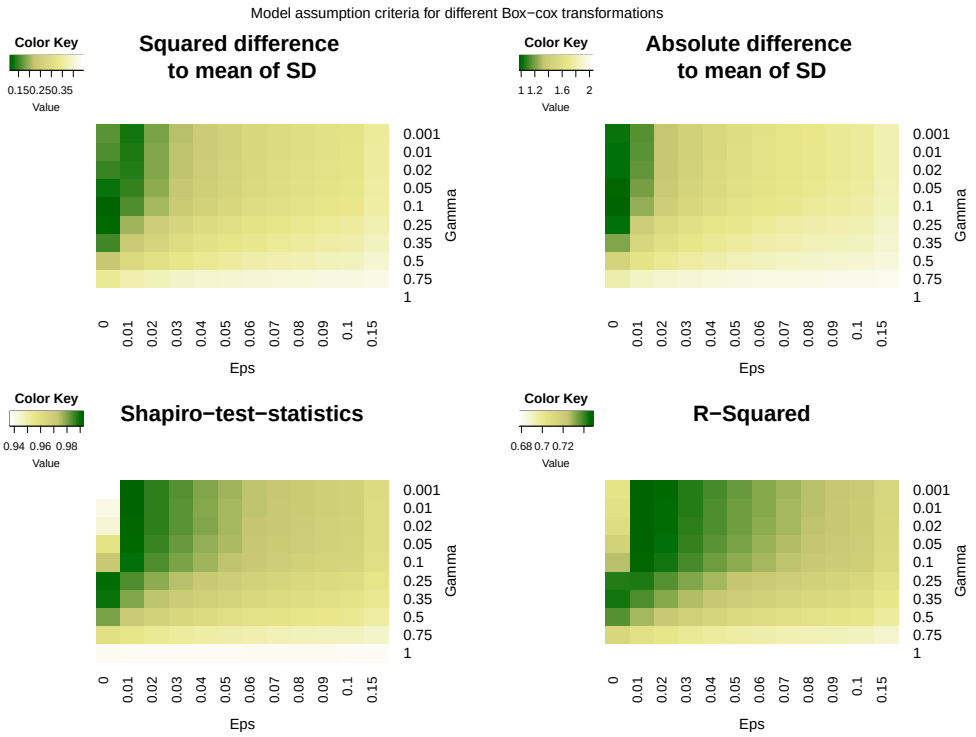

Figure 9: Comparison of model assumptions of Box-Cox transformed ρ under different values for γ and ϵ.

A.3 Bias Correction and Homoscedasticity Check

We applied a simulation based bias correction due to an observable bias especially for setups with small background rates. Therefore, recombination maps with in total 15 hotspots of lengths of 1kB (7) and 2kb (8) were simulated in a sequence of length 1000kb (1Mb). These recombination maps differ in 10 equidistant background rates between 0.001 and 0.011 with 15 replicates. The hotspots are between five and forty folds of the background recombination rates.

By estimating ρ with k = 1000 we use the systematic overlap of hotspot boundaries and segment boundaries to compare the estimator with the true value. This comparison (transformed scale) is provided in left plot of Figure 10 with a solid black diagonal line as perfect fit. Note that due to the overrepresentation of small recombination rates we have sampled as many background rates as hotspot rates in the recombination map. This yields approximately 4600 observations. We sampled the background rates uniformly from all background rates. Visual inspections reveal an overestimation of the background rate as well as an underestimation of very high ρ. A correction of these patterns is performed with quantile regressions where the estimated recombination rates explain the true recombination rates. The result of the estimated quantile regression for the 0.25 (orange), 0.35 (blue), 0.4 (green), and 0.5 (red) quantile, respectively, is given in Figure 10. On the right hand side of Figure 10 the residuals of the quantile regression models are plotted starting with the 0.25 quantile (top) and ending with 0.50 quantile (bottom).

Values smaller than -4 after bias-correction are set to -4, because they will equal to zero after the back-transformation.

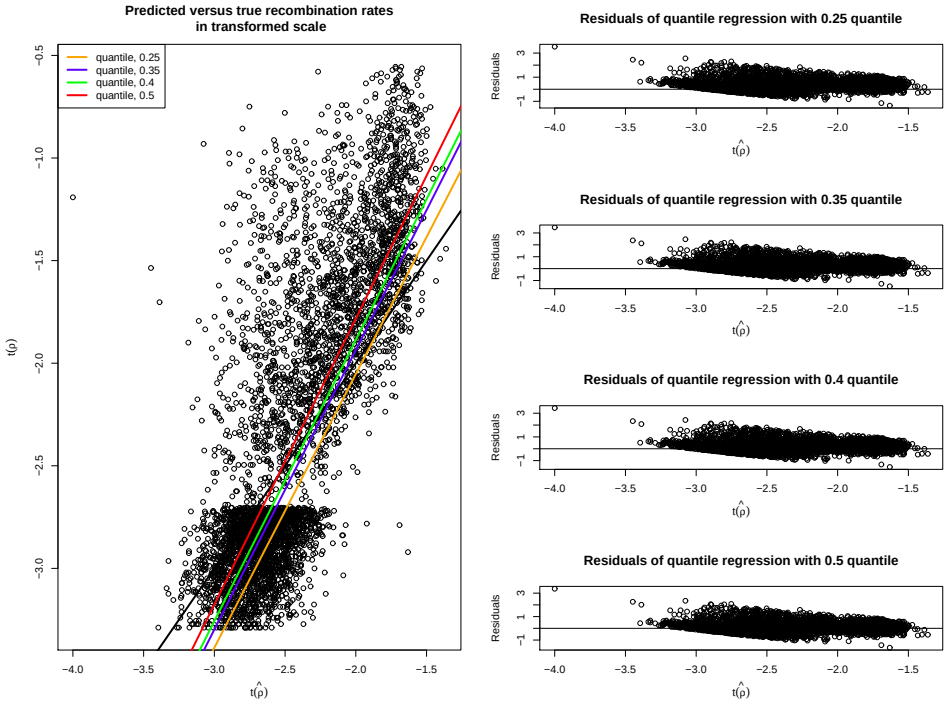

Figure 10: Left: Estimated versus true recombination rates based on recombination maps from simulations containing 15 hotspots of lengths 1 and 2kb. Predictions based on quantile regressions with 0.25 (orange), 0.35 (blue), 0.4 (green), and 0.5 (red) are added in this plot. Right: Residuals originating from the three quantile regressions provided for diagnostic purposes.

The SMUCE estimator requires homoscedastic observations [Frick et al., 2014]. Similar as to the approach in section S1.2 we analyze the homogeneity of the recombination rates by comparing the variance of the recombination rates in different intervals. Here, we divide the range of [0,0.2] in 25 equidistant segments. For each segment we compute the variance of the corrected (and back-transformed) recombination rates. By dividing the variance of each segment with the mean of all variances we have a measure of the variability of the variances along the considered recombination rates. In Figure 11 we shows ratios of variances divided by the mean of variances for all 24 considered intervals with an estimated smoothing spline for the four quantiles, 25% (left), 35% (middle-left), 40% (middle-right), and 50% (right). The difference of the variance to the mean variance only exceeds 20 percentage points in terms of variances for the first quartile in 3 intervals. When comparing the standard deviations (dashed lines) we can see that these deviations are less than (or slightly above) 10 percentage points (in absolute values) for (almost) all considered quantiles in the correction.

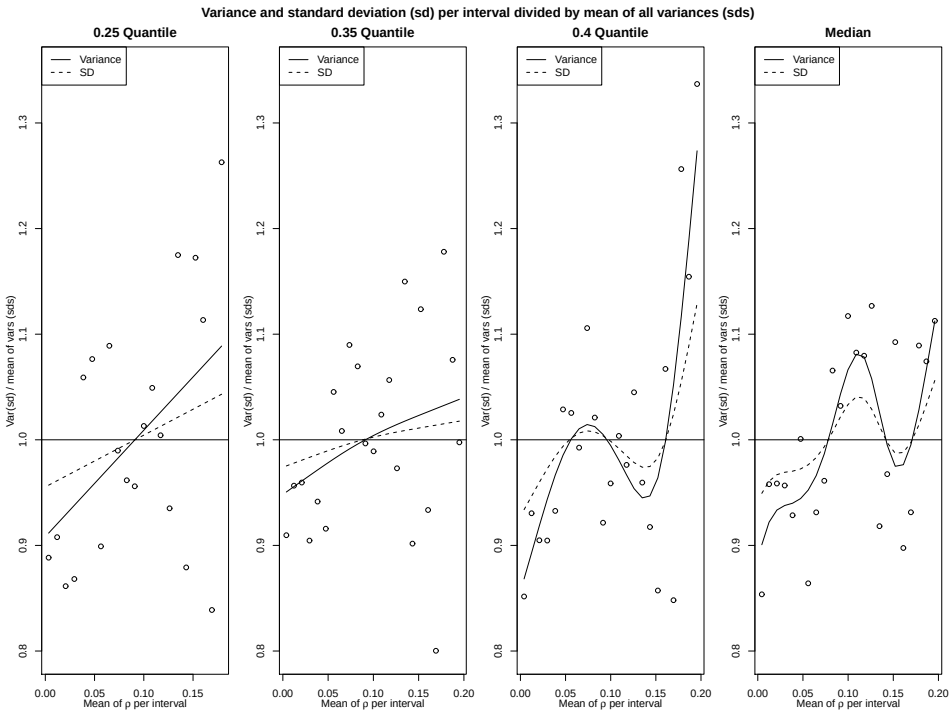

Figure 11: Graphical test for the homogeneity of the estimated recombination rates per quantile used in the quantile regression of the bias-correction (left: 0.25, left-middle. 0.35, right-middle: 0.4, right: 0.5). Variances are computed for 24 intervals of recombination rates between 0 and 0.2. The ratio of the variances divided by the overall mean of variances is plotted. The same approach is applied and visualized in terms of the standard deviation (dashed lines).

B Detailed Quality Assessment for Simple Setups

Figure 12 contains a more detailed analysis of the *simple* setups with boxplots including mean values as black dots accounting for sample sizes, recombination rates, and sequence lengths. We applied *LDJump* with 20 segments and a type I error probability of 5%. Hence, the considered segments had a length of 500 and 1000 (for 10kb and 20kb, respectively) nucleotides. Both sequence lengths have similar RMSE and are in the range of proper choices for segment lengths, see Figure 2. Especially for small to middle background rates (under the considered values) *LDJump* and *LDhat2* have on average a lower RMSE than *LDhat* and *LDhelmet*. *LDJump* and *LDhat2* have smaller RMSE for all considered sample sizes as well as sequence lengths, with slightly smaller values of our approach. Moreover, slightly smaller or equivalent RMSE was computed with *LDJump* for hotspot intensities from 5- to 20-fold the background recombination rate and hotspot lengths of 1/50 until 1/10. For a hotspot length of 1/5 of the total sequence length we can see a similar fit for all methods.

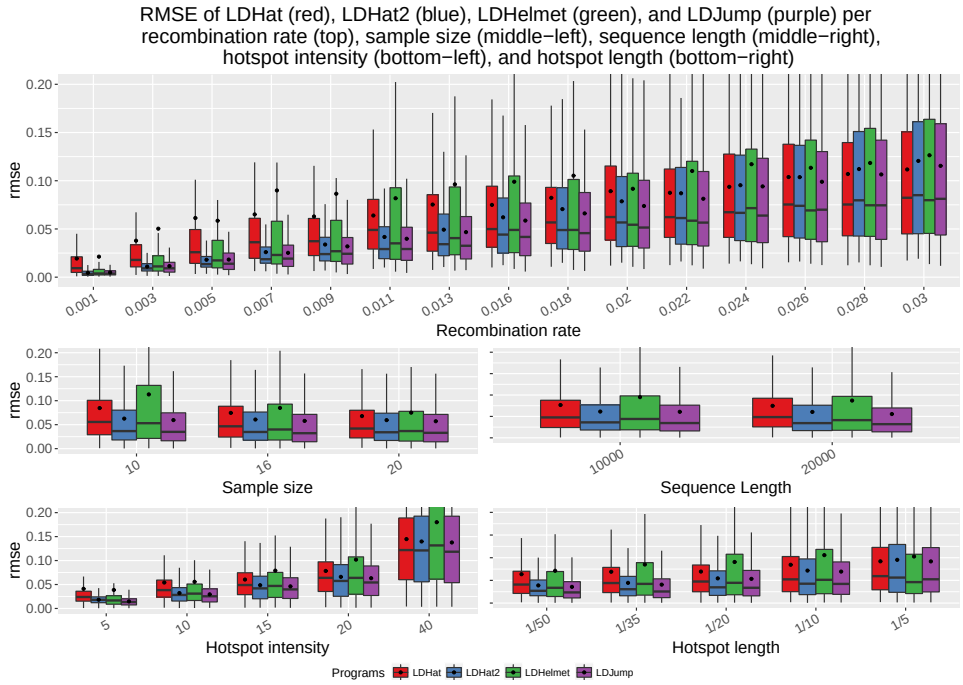

Figure 12: Comparison of the methods (*LDhat*(red), *LDhat2*(blue), *LDhelmet*(green), and *LDJump* (purple) with respect to the recombination rate (top), sample size (middle-left), sequence length (middle-right), hotspot intensity (bottom-left), and hotspot length (bottom-right).

C Runtime Comparison: Natural Setup

Table 7 shows average and median runtimes in seconds per 20 replicates of the 13 different *natural* setups. In the first five rows we provide the mean runtimes of *LDJump* with $k = 500$, 1000, 1500, and 2000, and of *LDhat*. The same pattern builds rows 6-10 for the median. The columns show the increasing background rates and highlight that the mean and (to a larger extent) the median of *LDhat2* is more strongly affected by larger recombination rates than *LDJump* with approximately constant runtimes across these setups. The runtime of *LDJump* is mainly determined by the computation of the summary statistics. However, *LDJump* has approximately (depending on the number of segments chosen) 20 times faster runtimes than *LDhat2*.

Runtime	Method	Background rates per base pair												
		0.01	0.013	0.022	0.027	0.039	0.045	0.054	0.062	0.071	0.08	0.085	0.091	0.1
Mean	500	6785	6766	6780	6725	6853	6768	6671	6805	6780	6684	6788	6716	6731
Mean	1000	4296	4288	4305	4262	4329	4286	4234	4303	4290	4234	4307	4257	4264
Mean	1500	3571	3571	3580	3561	3593	3616	3535	3566	3558	3584	3555	3628	3555
Mean	2000	3443	3468	3476	3448	3461	3508	3434	3452	3450	3458	3446	3511	3462
Mean	*LDhat2*	73902	73752	77171	87025	73832	74423	80707	70203	86679	81078	70239	74016	81053
Median	500	6825	6863	6809	6789	6865	6820	6758	6883	6811	6794	6833	6863	6785
Median	1000	4323	4355	4320	4315	4341	4320	4279	4353	4327	4307	4339	4355	4322
Median	1500	3577	3618	3594	3552	3644	3616	3546	3580	3624	3568	3603	3626	3568
Median	2000	3436	3482	3477	3458	3479	3464	3480	3453	3487	3430	3483	3458	3482
Median	*LDhat2*	100963	100963	124040	125921	100963	100934	125856	98629	126433	126072	98629	100963	126072

Table 7: Mean and median of runtime (in 1000s) per approach are provided. For $k = 1000, 1500, 2000$, and *LDhat* the runtimes in seconds are compared across all considered background recombination rates (columns).