OXFORD UNIVERSITY PRESS | Bioinformatics

popSTR: population-scale detection of STR variants

Journal:	*Bioinformatics*
Manuscript ID	BIOINF-2016-0522
Category:	Original Paper
Date Submitted by the Author:	04-Apr-2016
Complete List of Authors:	Kristmundsdóttir, Snædís; deCODE Genetics Inc, Statistics Sigurpálsdóttir, Brynja Dögg ; School of Science and Engineering, Reykjavík University, Engineering Kehr, Birte; deCODE genetics/Amgen, Statistics Halldórsson, Bjarni V.; deCODE genetics,
Keywords:	Bioinformatics, Mathematical modeling, Motif finding, Sequence analysis, Statistics, Algorithms

Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Advance Access Publication Date: Day Month Year
Manuscript Category

Genome analysis

popSTR: population-scale detection of STR variants

Snædís Kristmundsdóttir [1,*], Brynja D. Sigurpálsdóttir [2], Birte Kehr [1] and Bjarni V. Halldórsson [1,2,*]

[1]deCODE genetics/Amgen, Reykjavík, 101, Iceland and
[2]School of Science and Engineering, Reykjavík University, Reykjavík, 101, Iceland.

*To whom correspondence should be addressed.

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Motivation: Microsatellites, also known as short tandem repeats (STRs), are tracts of repetitive DNA sequences containing motifs ranging from 2-6 bases. The human reference genome contains approximately 1 million microsatellites, covering almost 1% of the genome (Gymrek *et al.*, 2016). Microsatellite analysis has a wide range of applications, including medical genetics, forensics and construction of genetic genealogy. However, microsatellite variations are rarely considered in whole-genome sequencing studies, in large due to a lack of tools capable of analyzing them (Duitama *et al.*, 2014).

Results: Here we present a microsatellite genotyper which is both faster and more accurate than other methods previously presented. There are two main ingredients to our improvements. First we reduce the amount of sequencing data necessary for creating microsatellite profiles by using previously aligned sequencing data. Second, we use population information to train microsatellite and individual specific error profiles. By comparing our genotyping results to genotypes generated by capillary electrophoresis we show that our error rates are 50% lower than those of lobSTR, another program specifically developed to determine microsatellite genotypes.

Availability: Source code is available on Github: `https://github.com/snaedis88/popSTR.git`

Contact: snaedis.kristmundsdottir@decode.is, bjarni.halldorsson@decode.is

1 Introduction

Microsatellites (a.k.a. short tandem repeats, STRs) are short DNA sequences containing a repeated motif of length 2-6 base pairs. Microsatellites are one of the most abundant type of variation in the human genome, after Single Nucleotide Polymorphisms (SNPs) and Indels. Microsatellites have a mutation rate estimated between $1 \cdot 10^{-4}$ and $1 \cdot 10^{-3}$ mutations per locus per generation (Sun *et al.*, 2012), much higher than the mutation rate estimated for SNPs (Kong *et al.*, 2012) of $1.2 \cdot 10^{-8}$. Due to their high mutation rate, the alleles of a microsatellite vary greatly between individuals (Sun *et al.*, 2012). Apart from identical twins, no pair of individuals alive today have the same combination of alleles for all microsatellites (Cox and Mays, 2000). Using relatively few microsatellites, it is possible to create a unique genetic profile for every individual (Cox and Mays, 2000), making microsatellites appealing for applications such as forensic analysis (Veselinović, 2006).

Their high mutation rate made microsatellites particularly alluring for genotyping during the linkage era (Gudbjartsson *et al.*, 2000). Despite their abundance and the increasing availability of whole genome sequencing data, microsatellites are however often neglected in GWAS studies (Gudbjartsson *et al.*, 2015).

The high mutation rate can be attributed to the repetitive structure of microsatellites, which causes a secondary DNA conformation that makes replication slippage events more likely than in other locations of the genome (Mirkin, 2007). Replication slippage occurs during DNA replication when the copy strand being created and the original template strand get shifted in their relative positions, causing a part of the template to either be copied twice or not copied at all (cf. Figure 1), resulting in either an increase or decrease in the number of motif repeats (Brown, 2002).

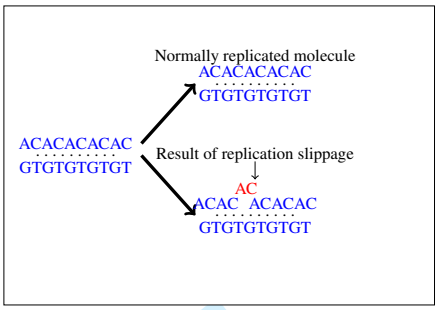

Fig. 1. An extra repeat element added because of replication slippage (Brown, 2002).

Replication slippage can occur within individual cells, as well as when the DNA sample is being analyzed. A slippage event that occurs during replication of a sex cell results in a germline mutation and may be passed on to an offspring, while slippage events within other cells of the body lead to somatic mutations. Slippage events also frequently occur in PCR amplification, a preprocessing step often performed prior to sequencing, or during sequencing itself. As a result, the sequence reads of an individual contain both reads from its germline variants and reads resulting from slippage events, complicating the genotyping of microsatellites.

The genotyping of microsatellites is further complicated by the fact that their high mutation rate can make it difficult to align microsatellite reads to the correct location on the genome; most popular read-to-reference aligners trade-off between the tolerance of insertions/deletions and running time. Yet another complication is the length of the microsatellite, as aligners generally require a unique match to the genome to seed their alignment, reads that are fully contained within a microsatellite can often not be placed within the genome. Further, reads that do not fully encompass the microsatellite and only contain a portion of the microsatellite can only give a lower bound on the number of repeats (Gymrek *et al.*, 2012).

A number of methods have been developed to genotype microsatellites (Gelfand *et al.*, 2014; Highnam *et al.*, 2013; Gymrek *et al.*, 2012). We present *popSTR*, a method capable of studying microsatellite (*STR*) variation within all individuals of a *pop*ulation simultaneously. Microsatellite mutation rates have been shown to vary greatly between microsatellites as well as between individuals (Sun *et al.*, 2012). Consequently, our model allows for an error model specific to each microsatellite and individual being studied.

Our results show that popSTR is both faster and more accurate than lobSTR (Gymrek *et al.*, 2012), a previously described method for determining microsatellites. popSTR also finds more microsatellite genotypes than the general purpose genotype caller GATK (McKenna *et al.*, 2010), with the ones found also being more reliable.

2 Methods

popSTR requires three inputs; a reference genome, a list of microsatellite locations (markers) on the reference genome and sequencing data of the set of individuals (population) being studied. We assume that the sequencing data is Illumina whole genome paired-end sequencing data, mapped to the reference genome and stored in BAM-files, with one BAM file per individual. The output of popSTR are; for each marker the set of alleles occurring in at least one individual in the population and the genotype likelihoods of all allele pairs of the marker for each individual.

popSTR starts by determining a set of informative reads for each marker/individual pair and computing various attributes for the reads. Subsequently, an iterative algorithm is employed to train error models and report genotypes.

2.1 Read selection and processing

The input to the read selection algorithm is a BAM-file, containing the read pairs of a single individual, j, the reference genome and a file containing a set, I, of microsatellite locations. The algorithm outputs for each microsatellite $i \in I$, a set R_{ij} of reads aligned to the microsatellite and for each read $r \in R_{ij}$ a set of attributes computed for r.

The algorithm iterates through the sequencing data and the microsatellite location file in parallel and compares read coordinates to microsatellite coordinates. For each microsatellite, i, we determine a set of candidate informative reads as those reads whose alignment intersects the microsatellite location as well as unmapped mates of reads that have been mapped near the microsatellite (within a fixed distance, chosen by default as 1000 bp).

For each candidate informative read we first determine if the read contains the repeat motif of the microsatellite. Those reads that contain the repeat sequence are aligned to the sequences flanking the microsatellite location. The read is split into three parts; the sequence before the microsatellite, the microsatellite repeat sequence and the sequence after the microsatellite. Figure 2 shows how two subsequences are constructed from the read, containing the repeat and the flanking base pairs on either side. Both subsequences are aligned to the reference genome using an overlap alignment and the Needleman-Wunsch algorithm; the first sequence is aligned to the bases preceding the repeat in the reference and the second is aligned to the bases following the repeat in the reference. If the sum of the alignment scores exceeds a minimum threshold the read is considered aligned. The user also specifies a minimum number of flanking bases needed on each side of the repeat. Aligned reads that meet this threshold are added to R_{ij}.

To increase our sensitivity in identifying microsatellite containing reads we also process reads when there is a strong support for the alignment on one side of the microsatellite, while only few bases can be aligned at the other end. We also consider reads to be aligned at both ends if at least four bases can be aligned on each side. Such reads are added to R_{ij} if the sum of the aligned flanking bases is greater than or equal to twice the user specified minimum number of flanking bases on each side.

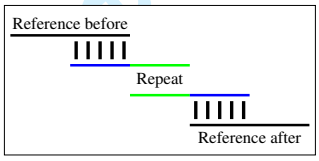

Fig. 2. We split the read into two overlapping parts where the first part has the repeat as a suffix and the second part has the repeat as a prefix. We then align the first part to the reference sequence preceding the microsatellite and the second part to the reference sequence after the microsatellite.

We estimate the length, $l^i(r)$ of the microsatellite repeat in r as the number of bases in r between the last base aligned to the sequence preceding i in the reference and the first base aligned to the sequence following i in the reference. We represent alleles of a microsatellite i with the number of times its repeat motif $m(i)$ is repeated. We let $|m(i)|$ represent the length of the repeat motif of microsatellite i, the allele A_r

reported by r can be computed as:

$$A_r = \frac{l^i(r)}{|m(i)|} \quad (1)$$

Some microsatellite alleles are very long, at times longer than the read length used for sequencing. Reads overlapping long microsatellites can only give partial information on the length of the microsatellite allele; the length of the microsatellite allele must be at least as long as the overlap of the read with the microsatellite. To address the challenge presented by reads only able to give a lower bound on the number of repeats, we set a user-specified maximum length of allele, m_l. All alleles longer than m_l are lumped together and reported as the composite allele $\geq m_l$. Reads that contain repeats that span the entire length of the read or occur at either end of a read and the base pair length of the repeat is at least m_l are processed and the number of repeats is reported as:

$$A_r = \frac{m_l}{|m(i)|} \quad (2)$$

For each read $r \in R_{ij}$ we store a number of attributes relevant to the alignment, summarized and defined in Table 2. These attributes were chosen with the intent of revealing reads that are the result of misclassification events, i.e. sequencing or mapping errors.

2.2 Iterative algorithm

There are multiple sources of error that need to be accounted for in our model. Replication slippage is dependent on the marker being considered as well as the individual. In addition, some of the reads may be the results of sequencing or mapping errors.

Replication slippage has two forms; *full motif slippage* and *stutter noise*. A *full motif slippage* is when the length of the slippage is an integer multiple of the length of the repeat motif of the microsatellite, all other slippages are referred to as *stutter noise*. Following lobSTR (Gymrek *et al.*, 2012) we model these two types of slippage events separately. We assume a Poisson distribution for full motif slippage events and a geometric distribution for stutter noise. In what follows, we will refer to the rate of full motive slippage events as *slippage rate* while we will refer to the rate of stutter noise as *stutter rate*.

Sequencing and mapping errors are accounted for using logistic regression classification of the reads for each microsatellite separately. Based on the attributes computed above and the genotype of an individual at the microsatellite, the classifier assigns a probability to each read of being an error read, i.e. the result of a mapping or sequencing error.

We use an iterative approach to simultaneously train logistic regression classifiers, estimate slippage and stutter rates for each microsatellite and a slippage rate for each individual. We start by describing the individual steps of our algorithm and then show how these are combined into an algorithm.

2.2.1 Read classification

To identify reads resulting from sequencing or mapping errors we train a logistic regression classifier (Hosmer Jr and Lemeshow, 2004) for each microsatellite using the reads of all individuals. At each iteration of the algorithm, each individual has a currently estimated genotype at the microsatellite. This currently estimated genotype allows us to label reads as either TRUE or FALSE. Reads reporting one of the two alleles in the current genotype are labelled as TRUE and reads reporting other alleles, that further cannot be explained with a single slippage event, are labelled as FALSE. We use the attributes computed in the read selection step (cf. Table 2) as control variables for the logistic regression classifiers.

The resulting classifier allows us to assign a probability, $p_i(r)$, to each read, r, representing the probability that r is correctly classified as a read from microsatellite i. Reads classified as TRUE are believed to represent the sequence of the individual at the marker being considered. Reads classified as FALSE are believed to be the result of a mapping or a sequencing error.

2.2.2 Slippage rate estimation

The frequency of slippage events varies between microsatellites. To account for this we estimate a marker specific slippage rate.

Assuming we know which reads are the results of a full motif slippage event, we can estimate the slippage rate at microsatellite i by dividing the number of reads resulting from full motif slippage by the total number of reads aligned to the microsatellite. S_i^M, the slippage rate at microsatellite i, could be estimated as:

$$S_i^M = \frac{n_i^!}{n_i} \quad (3)$$

where $n_i^!$ represents the number of reads aligned to microsatellite i that do not support the current genotype and are considered to be results of a full motif slippage and n_i represents the total number of reads aligned to microsatellite i.

The above expression however ignores the fact that individuals may have different slippage rates. We assume that the slippage of marker i in individual j, S_{ij}, is a composite of a marker specific slippage rate, S_i^M, and an individual specific slippage rate S_j^P.

$$S_{ij} = S_i^M + S_j^P \quad (4)$$

Given the current genotype of individual j at marker i we construct the set $R_{ij}^!$ of those reads that do not agree with either of the alleles of the current genotype and are considered to be the result of full motif slippage events.

We can then estimate S_{ij} as:

$$S_{ij} = \frac{\sum_{r \in R_{ij}^!} p_i(r)}{\sum_{r \in R_{ij}} p_i(r)} \quad (5)$$

Consequently, we can estimate S_i^M and S_j^P as

$$S_i^M = \frac{\sum_{r \in R_{ij}^!} p_i(r)}{\sum_{r \in R_{ij}} p_i(r)} - S_j^P \quad (6)$$

$$S_j^P = \frac{\sum_{r \in R_{ij}^!} p_i(r)}{\sum_{r \in R_{ij}} p_i(r)} - S_i^M \quad (7)$$

Giving us multiple estimates for each S_i^M and S_j^P. We weight these estimates by the inverse variance of S_{ij} and the number of correctly classified reads at microsatellite i in individual j. The variance of S_{ij} is $S_{ij}(1 - S_{ij})$, assuming S_{ij} obeys a binomial distribution. The weight, w_{ij} of microsatellite i in individual j is then:

$$w_{ij} = \frac{\sum_{r \in R_{ij}} p_i(r)}{S_{ij}(1 - S_{ij})} \quad (8)$$

Allowing us to estimate S_i^M and S_j^P as:

$$S_i^M = \sum_j \frac{w_{ij}}{\sum_j w_{ij}} \cdot \left(\frac{\sum_{r \in R_{ij}^!} p_i(r)}{\sum_{r \in R_{ij}} p_i(r)} - S_j^P \right) \quad (9)$$

$$S_j^P = \sum_i \frac{w_{ij}}{\sum_i w_{ij}} \cdot \left(\frac{\sum_{r \in R_{ij}^!} p_i(r)}{\sum_{r \in R_{ij}} p_i(r)} - S_i^M \right) \quad (10)$$

2.2.3 Stutter rate estimation

Following the model presented in lobSTR (Gymrek *et al.*, 2012), we estimate a microsatellite specific parameter t_i, for the geometric distribution assumed for stutter noise as:

$$t_i = \frac{1}{1+\bar{x}_i} \tag{11}$$

Where $\bar{x}_i$ is an estimate of the fraction of reads at microsatellite i that are results of stutter noise events.

To estimate $\bar{x}_i$ we start by computing the absolute value of the minimum base pair distance to the current genotype for all reads covering microsatellite i. A read from individual j, supporting either allele of the individual's current genotype (A,B) has a distance of zero but reads not supporting the current genotype have a distance of:

$$dist(r) = min(|l(A) - l^i(r)|, |l(B) - l^i(r)|) \tag{12}$$

where $l(A)$ and $l(B)$ represent the base pair length of alleles A and B, respectively and $l^i(r)$ represents the base pair length of the allele reported by the read. We then estimate $\bar{x}_i$ as the average of this number modulo the length of the repeat motif at microsatellite i.

2.2.4 Computing genotype likelihoods

We focus our attention on determining the likelihood of a genotype, gt. We are given a set R of reads, which we assume are independent observations of the microsatellite i, allowing us conclude that:

$$L(R|gt) = \prod_{r \in R} L(r|gt) \tag{13}$$

We now show how to compute $L(r|gt)$, adding terms for each source of error successively to our model. We first consider the case when the only sources of error are full motif slippage events and read misclassification events. Recall, that A_r represents the number times the repeat motif of i is repeated in r and that the alleles of a genotype are represented with the number of times the repeat motif, m_i, is repeated. Given an allele, A, we compute $x_r(A)$ as the number of slippage events needed to explain r with A as $x_r(A) = |A - A_r|$. We assume that the number of slippage events follows a Poisson distribution with $\lambda = S_{ij}$. This gives the following expression for a homozygous genotype $gt = (A, A)$.

$$L(r|A, A) = p_i(r) \cdot pois(x_r(A); S_{ij}) \tag{14}$$

For a heterozygous genotype (A, B) we assume that each allele is drawn with equal probability:

$$L(r|A, B) = \\ p_i(r) \cdot \left(\frac{1}{2} \cdot pois(x_r(A); S_{ij}) + \frac{1}{2} \cdot pois(x_r(B); S_{ij})\right) \tag{15}$$

The above expression assigns a very small likelihood for reads that are not the results of slippage events. With probability $1 - p_i(r)$ the read being considered is an error read, in this case we assume that each of the other reported alleles is equally likely. We let n^i be the number of alleles present in the population for microsatellite i and refine our expression for $L(r|A, B)$ as follows:

$$L(r|A, B) = p_i(r) \cdot \left(\frac{1}{2} \cdot pois(x_r(A); S_{ij}) + \\ \frac{1}{2} \cdot pois(x_r(B); S_{ij})\right) + \frac{1 - p_i(r)}{n^i} \tag{16}$$

Slippage events are more likely to delete repeat units than insert. To account for this, we further refine our model and add a parameter, p_d, representing the probability that if a slippage event occurs, this event results in a deletion of a motif. Given an allele A and a read r we compute a_r^A as p_d if $A - A_r \leq 0$ and $1 - p_d$ if $A - A_r > 0$. Our refined model then becomes:

$$L(r|A, B) = p_i(r) \cdot \\ \left(\frac{1}{2} \cdot pois(x_r(A); S_{ij}) \cdot a_r^A + \frac{1}{2} \cdot pois(x_r(B); S_{ij}) \cdot a_r^B\right) \\ + \frac{1 - p_i(r)}{n^i} \tag{17}$$

Finally, we account for stutter noise, for which we assume a geometric distribution and use the marker specific t_is estimated using Equation 11. To reflect this in our model we split $x_r(A)$ and $x_r(B)$ into their integer and decimal portions. We let $x_r^k(A)$ denote the integer portion and $x_r^d(A)$ the decimal portion of $x_r(A)$. Similarly we split $x_r(B)$ into $x_r^k(B)$ and $x_r^d(B)$ and our final model becomes:

$$L(r|A, B) = p_i(r) \cdot \\ \left(\frac{1}{2} \cdot pois(x_r^k(A); S_{ij}) \cdot geom(x_r^d(A); t_i) \cdot a_r^A \right. \\ \left. + \frac{1}{2} \cdot pois(x_r^k(B); S_{ij}) \cdot geom(x_r^d(B); t_i) \cdot a_r^B\right) \\ + \frac{1 - p_i(r)}{n^i} \tag{18}$$

Given a set of reads $R_{i,j}$ for a microsatellite i and individual j we compute this genotype likelihood for all genotypes A, B present in the population. The *current genotype* is the A, B with the highest $\prod_{r \in R_{i,j}} L(r|A, B)$.

2.2.5 Algorithm pseudocode

The algorithm can now be described with the following pseudocode:

- Select and process reads
- Initialize genotypes.
- Initialize all S_i^M, S_j^P, p_i, t_i.
- While genotypes have not converged:
  - Use S_i^M, p_i, t_i, S_j^P to compute genotypes.
  - Update S_j^Ps using S_i^Ms,p_is and t_is.
  - From the current genotypes determine which reads are TRUE and FALSE.
  - Update p_is using read classification.
  - Update t_is using current genotypes.
  - Update S_i^Ms using current genotypes, S_j^Ps and p_is.
- Compute genotype likelihoods and exit.

2.3 Kernelization of iterative algorithm

Our iterative algorithm can be too memory and time intensive for large data sets. In order to make our time and memory requirements more manageable we can kernelize our algorithm. We select a small set of well behaving microsatellites and individuals with high quality sequencing data for our initial training, a set we refer to as a *kernel*. Within this kernel we apply the full algorithm described in 2.2.5.

Once this kernel has been trained we estimate individual specific slippage rates, S_j^Ps, using only the markers within the kernel, keeping the marker slippage rates, S_i^Ms, the stutter rates, t_is, and the marker classification models ($p_i(r)$s) fixed.

Once the S_j^Ps have been trained for all individuals, j, we train S_i^M, t_i and $p_i(r)$ for all markers i keeping the S_j^Ps fixed, allowing us to compute a final set of genotype likelihoods.

3 Implementation

PopSTR was written in C++ using the sequence analysis library SeqAn (Döring *et al.*, 2008) which allows for easy reading and manipulation of data stored in BAM-files.

The implementation of popSTR has four steps. In the first step we identify the reads useful for genotyping, compute their attributes and initialize genotypes. In the second step we estimate S_j^Ps, S_i^Ms, t_is and p_is on a kernel of markers. In the third step we use results from the kernelization to compute S_j^Ps. In the final step we train S_i^Ms, t_is and p_is and finally perform genotyping.

3.1 Read selection and processing

We use the fact that the sequencing data has already been aligned (in a BAM file), allowing us to limit the number of reads that we consider. We can however not limit our search only to reads that have been aligned to a microsatellite, as alignment to microsatellites by general purpose aligners, such as BWA (Li and Durbin, 2009), is not reliable. General purpose aligners trade accuracy and speed in their implementation and do not account for the high mutation rate of microsatellites. We limit our search to reads that have been aligned to microsatellites and reads that are unaligned but have a mate that is aligned near the microsatellite being considered. Sequences already aligned to non-microsatellite sequences are unlikely to be useful while sequences that are unaligned may in fact contain a microsatellite but have not been aligned because they are too different from the reference.

When selecting reads and in order to perform the read classification we compute a number of attributes related to the reads' alignment and their sequencing quality. As previously mentioned, candidate microsatellite reads are processed by first identifying the repeat sequence within the read. Subsequently, the sequences flanking the repeat are aligned to the sequences flanking the microsatellite in the reference genome. The quality of this alignment is one of the attributes used as a control variable in the logistic regression classification. We define *purity* of an alignment as the number matching base pairs divided by the total number of base pairs in the alignment. The purity of a microsatellite repeat sequence is the number of base pairs matching the repeat divided by the total number of base pairs in the repeat. The purity of the repeat sequence in the read is another control variable. All attributes computed, used as control variables in the logistic regression, are summarized in Table 2.

Further, some attributes are required to reach a minimum value for the read to be used. The minimum microsatellite purity required is relative to the purity of the microsatellite sequence in the reference and also depends on the number of flanking bases available in the read. Table 1 summarizes these filters used.

Table 1. Minimum numeric values when identifying useful reads

Name	Condition	Minimum value
Microsatellite purity	both flanking	0.75*(ref. purity)
	one sided flanking	0.8*(ref. purity)
	no flanking	0.85*(ref. purity)
Alignment purity	one sided flanking	0.7
# repeats	motiflength = 2	4
	motiflength = 3	3
	motiflength $\in 4, 5, 6$	2

Finally, we do not consider low quality reads, i.e. the ones that fail platform or vendor quality checks nor reads that are PCR or optical duplicates.

3.2 Kernelization

Convergence has been reached in the kernelization when less than 0.5% of the genotypes are updated between iterations.

We initialize the slippage rate for individual j, using the following expression

$$S_j^P = \frac{n_j^!}{n_j} \tag{19}$$

where $n_j^!$ represents the number of reads from individual j not supporting the initialized genotype and n_j represents the total number of reads from individual j.

3.3 Individual slippage rate computation

The marker slippage and stutter rates estimated (S_i^Ms and t_is) and the logistic regression classifiers ($p_i(r)$s) trained during the kernelization are used to directly estimate the individual specific slippage rates (S_j^Ps). First, we compute the attributes of reads aligned to the microsatellites in the kernel. Next, we assign misclassification probabilities, $p_i(r)$s to the reads using the logistic regression classifiers from the kernel and we update the genotypes, with marker slippage and stutter rates from the kernel using the expression given in Equation 18 to determine the most likely genotype. Finally, we use the expression given in Equation 10 to estimate an individual slippage rate, S_j^Ps. We iterate this process, keeping the marker specific properties from the kernel constant, until the individual slippage rates, (S_j^Ps), have reached convergence.

3.4 Marker slippage and stutter rate computation, logistic regression and genotyping

We fix the probability of deletion Equation 18 as $p_d = 0.85$ and consequently $1 - p_d = 0.15$. We iterate between updating genotypes and updating the microsatellite slippage and stutter rates, (S_i^Ms and t_is), and logistic regression classifiers ($p_i(r)$s), while keeping individual slippage rates, (S_j^Ps), constant, until convergence has been reached.

4 Results

4.1 Data set

We analyzed microsatellites for 15,220 whole genome sequenced individuals, sequenced using Illumina sequencers. Sequencing reads had previously been mapped to GRCh38 using BWA (Li and Durbin, 2009).

We ran Tandem Repeat Finder to determine the microsatellite locations (Benson, 1999). This resulted in a set of 880,355 microsatellite locations found on GRCh38 autosomes after excluding locations within 100bp of each other and microsatellites located in high coverage regions.

We initially ran popSTR on a set of 8,453 individuals and 880,355 microsatellites. We chose a kernel set of 703 individuals with high quality sequencing data and 8,303 microsatellites on chromosome 1, based on their imputation info (Gudbjartsson *et al.*, 2015) when imputed into the Icelandic population.

Our comparison set is based on the genotypes of 15,220 individuals on 880,355 microsatellites. Out of these a total of 380,261 microsatellites were found to be polymorphic and were subsequently imputed into the Icelandic population (Gudbjartsson *et al.*, 2015).

For comparison purposes, we chose 141 markers where capillary electrophoresis benchmark genotypes were available, sequenced as parts

Table 2. The attributes used as control variables in the Logistic regression classification.

Attribute	Definition
Quality score	Mapping quality score of the aligned read.
Microsatellite Purity	# of base pairs matching microsatellite repeat sequence/ # of base pairs in microsatellite sequence
Repeat bases over 20	The number of base pairs with a PHRED-scaled quality over 20 in the microsatellite sequence.
Flanking bases on right over 20	The number of base pairs with a PHRED-scaled quality over 20 in flanking bases after the repeat.
Edit distance of mate	Edit distance of aligned base pairs of the mate sequence to the reference.
Left side alignment score	Alignment score of sequence before the microsatellite to the reference.
Right side alignment score	Alignment score of sequence after the microsatellite to the reference.
Was unaligned	Boolean value indicating if the read was unaligned by BWA.
Alignment shift	Measures changes from original alignment during the realignment of flanking sequences.
Read length	Total length of the read.

of various disease association efforts at deCODE genetics (Sun *et al.*, 2012).

Comparisons to lobSTR were done by choosing 10 individuals from the 15,220 sequenced individuals. The 10 individuals were chosen to have a large number of electrophoresis genotypes available.

The 15,220 samples were also genotyped using the GATK (McKenna *et al.*, 2010) genotype caller and imputed into the Icelandic population. GATK is a general purpose genotype caller that does not distinguish between indels and microsatellites. To further investigate the quality of our genotypes we matched our microsatellites coordinates to output coordinates of indel alleles from the GATK genotype caller where the indel allele matched the microsatellite repeat motif. We then compared the imputation results into the Icelandic population for markers where a match was found.

4.2 Comparison to lobSTR

We compared the popSTR and lobSTR genotypes to inhouse benchmark data obtained through capillary electrophoresis. The capillary electrophoresis genotypes are represented as base pair distances from a reference individual, while genotypes reported from sequencing (by popSTR or lobSTR) are presented as lengths of the microsatellite alleles. As we did not have the length of the microsatellite alleles of the reference individual, we considered the genotypes reported from capillary electrophoresis and sequencing to agree if an identical difference in lengths between the two alleles carried by the individual was reported by the two methods.

Both lobSTR and popSTR can be expected to report more accurate genotypes when more reads overlapping the microsatellite are used in the genotyping. We therefore condition our results on the number of reads used in the genotyping. Figure 3 shows the accuracy of lobSTR and popSTR as a function of the number of reads used by lobSTR. The figure clearly shows that, as expected, the accuracy of both methods increases when more reads overlapping the microsatellite are used. The figure also shows that popSTR consistently has higher genotyping accuracy than lobSTR.

Table 3 summarizes the comparison between the two methods. We observe that, when we restrict our analysis to microsatellites and individuals where there are at least 10 reads overlapping the microsatellite, popSTR has a 96% agreement with the capilllary electrophoresis genotypes while lobSTR has a 92% agreement. Consistently, over all coverage thresholds the number of genotypes that are in disagreement with the capillary electrophoresis genotypes is approximately 2 times higher for lobSTR than for popSTR, i.e. the error rate of popSTR is 50% lower than that of lobSTR.

Table 3. Genotyping accuracy of lobSTR and popSTR compared to capilllary electrophoresis genotypes. The results are thresholded on the number of reads available to lobSTR.

Coverage filter	lobSTR	popSTR
>=1	87.3%	93.5%
>=5	89.5%	94.3%
>=10	92.0%	96.0%
>=15	93.5%	96.4%
>=20	94.4%	97.2%

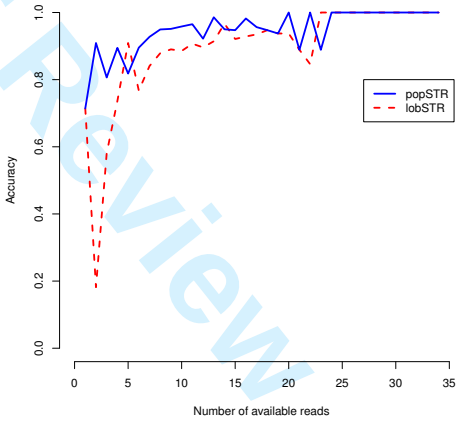

Fig. 3. The accuracy of the lobSTR and popSTR genotypers as a function of the number of reads used by lobSTR overlapping the microsatellite. Results are averages over 10 individuals and 141 microsatellites.

To further confirm the accuracy of our method we compared the popSTR genotypes of 409 individuals to the benchmark genotypes, considering the same 141 markers as in the comparison to lobSTR. Figure 4 shows how the accuracy of popSTR increases with the number of reads used in the genotyping.

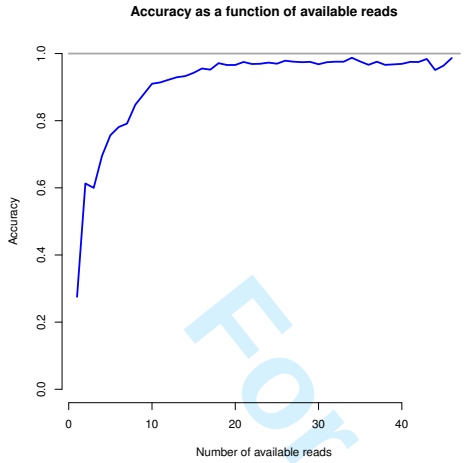

Fig. 4. The accuracy of the popSTR genotyper as a function of the number of reads used by popSTR overlapping a microsatellite. Results are averages over 409 individuals and 141 microsatellites.

We compared the running times of popSTR and lobSTR and found an average speed-up provided by popSTR of 74.7%. The average running time per individual was 39.2 hours (s.d. 6.7 hours). This includes the time of both the aligning and allelotyping steps of lobSTR. Since the genotyping step of popSTR was performed for all individuals simultaneously we report the total running time of this step divided by the number of individuals (15,220). A similar average is reported for the running time of the kernelization. Other steps of popSTR are performed per sample. Summing up the running time of all steps of popSTR we get a total of 9.9 hours per individual.

4.3 Comparisons to GATK

We use imputation info (Gudbjartsson *et al.*, 2015) to compare the quality of genotypes reported by GATK and popSTR. Imputation info is a measure between 0 and 1, representing confidence in genotype assignment reported by the imputation software (Gudbjartsson *et al.*, 2015), with larger values of imputation info representing higher confidence. We have previously determined imputation info of greater than or equal to 0.9 as a threshold for which we believe that the genotypes are highly reliable (Gudbjartsson *et al.*, 2015).

GATK is a general purpose tool for determining genotypes and does not have a specific model for microsatellites, but rather lumps them in a category with indels. We compared the imputation info of popSTR microsatellites to the imputation info of indel alleles from GATK in cases where alleles reported by GATK were located within a microsatellite sequence. At microsatellite locations, some of the indels reported by GATK contain the microsatellite motif, while others do not. We condition our comparison to GATK on whether the microsatellite motif is found in the indel reported by GATK.

For a judicious comparison, we construct a single number for each microsatellite by summing the info of each allele weighted by frequency. This is shown in Equation 20 where i_w represents the weighted info value and f_a and i_a represent the frequency and imputation info of allele a, respectively.

$$i_w = \frac{\sum_a f_a * i_a}{\sum_a f_a} \tag{20}$$

For a total of 152,152 microsatellites found by popSTR an indel was reported by GATK within the microsatellite. In 107,104 or 70.4% of those microsatellites the imputation info of popSTR was higher than that of GATK. The number of microsatellites where the imputation info of either popSTR or GATK was above 0.9 was 133,366. In 99,787 (74.8% of 133,366) the imputation info was higher for popSTR than GATK. popSTR had imputation info greater than 0.9 for 120,317 or 79.7% of the microsatellites found by both methods and GATK for 92,854 or 61.0% of them. When either GATK or popSTR had imputation info greater then 0.9 the average of popSTR was 0.95 (s.d. 0.16) and the average of GATK was 0.9 (s.d. 0.16).

In a 75,057 of the microsatellites found by popSTR the indel reported by GATK contained the microsatellite motif. In 56,521 (75.3%) the imputation info of popSTR was higher than that of GATK. There were 68,216 microsatellites where the imputation info of either popSTR or GATK was greater than 0.9 and for 53,812 or 78.9% of those the imputation info of popSTR was higher than that of GATK. For 62,962 of the 75,057 microsatellites (83.9%), the imputation info of popSTR was greater than 0.9 and for 49,684 (66.2%) the imputation info of GATK was greater than 0.9. The average imputation info, restricted to the set of microsatellites when either popSTR or GATK had imputation info greater than 0.9, was 0.96 (s.d 0.12) for popSTR and 0.93 (s.d. 0.08) for GATK.

4.4 Individual slippage rate as a function of age

Individual slippage rates, (S_j^Ps), can be expected to increase with age due to somatic replication slippage mutations that accumulate during a person's life. We tested this hypothesis, restricting our analysis to 12,084 individuals that were sequenced using Illumina HiSeqX sequencing machines. After correcting for the sequencing protocols used, we observed a highly significant (p-value: $8.2 \cdot 10^{-13}$) increase in slippage rate with age. This result is a further justification of our decision to include individual specific slippage rates in our model and suggests that they are measuring a biologically meaningful attribute.

5 Conclusion

Here we have shown that, by creating a microsatellite profile for an individual using previously aligned data, it is possible to significantly decrease the running time of microsatellite genotyping by considering only reads that are either aligned to a known microsatellite location or not aligned at all. The filtering dismisses a large portion of the data immediately while minimally effecting the microsatellite profile. Our results also show that the genotyping accuracy of our program is higher than for the general purpose genotype caller GATK as well as lobSTR, a program specifically designed for calling of microsatellites.

Several improvements could still be made to our model and method. Our method does not consider reads where neither the read nor its mate align to the reference genome. Our method also assumes that the mate of the read containing the microsatellite is correctly mapped. If the read pair were to be mapped to a graph reference (a reference genome containing all variants) it is possible that a joint alignment of both the read containing the microsatellite and its mate would reveal the correct location for the read pair. We do not account for possible sampling biases, i.e. it may be more likely that we observe reads that are similar to the reference than those that are highly divergent from the reference. Similarly, there may be biases introduced by our alignment algorithm or filtering steps not accounted for in our model.

References

Benson, G. (1999). Tandem repeats finder: a program to analyze dna sequences. *Nucleic Acids Res*, **27**(2), 573–580.

Brown, T. A. (2002). *Genomes*. Wiley-Liss, Oxford, 2nd edition.

Cox, M. and Mays, S. (2000). *Human osteology: in archaeology and forensic science*. Cambridge University Press.

Döring, A., Weese, D., Rausch, T., and Reinert, K. (2008). SeqAn an efficient, generic c++ library for sequence analysis. *BMC Bioinformatics*, **9**, 11.

Duitama, J., Zablotskaya, A., Gemayel, R., Jansen, A., Belet, S., Vermeesch, J. R., Verstrepen, K. J., and Froyen, G. (2014). Large-scale analysis of tandem repeat variability in the human genome. *Nucleic Acids Res*.

Gelfand, Y., Hernandez, Y., Loving, J., and Benson, G. (2014). VNTRseek-a computational tool to detect tandem repeat variants in high-throughput sequencing data. *Nucleic Acids Res*, **42**(14), 8884–8894.

Gudbjartsson, D. F., Jonasson, K., Frigge, M. L., and Kong, A. (2000). Allegro, a new computer program for multipoint linkage analysis. *Nat Genet*, **25**(1), 12–13.

Gudbjartsson, D. F., Helgason, H., Gudjonsson, S. A., Zink, F., Oddson, A., Gylfason, A., Besenbacher, S., Magnusson, G., Halldorsson, B. V., Hjartarson, E., Sigurdsson, G. T., Stacey, S. N., Frigge, M. L., Holm, H., Saemundsdottir, J., Helgadottir, H. T., Johannsdottir, H., Sigfusson, G., Thorgeirsson, G., Sverrisson, J. T., Gretarsdottir, S., Walters, G. B., Rafnar, T., Thjodleifsson, B., Bjornsson, E. S., Olafsson, S., Thorarinsdottir, H., Steingrimsdottir, T., Gudmundsdottir, T. S., Theodors, A., Jonasson, J. G., Sigurdsson, A., Bjornsdottir, G., Jonsson, J. J., Thorarensen, O., Ludvigsson, P., Gudbjartsson, H., Eyjolfsson, G. I., Sigurdardottir, O., Olafsson, I., Arnar, D. O., Magnusson, O. T., Kong, A., Masson, G., Thorsteinsdottir, U., Helgason, A., Sulem, P., and Stefansson, K. (2015). Large-scale whole-genome sequencing of the icelandic population. *Nat Genet*, **47**(5), 435–444.

Gymrek, M., Golan, D., Rosset, S., and Erlich, Y. (2012). lobstr: A short tandem repeat profiler for personal genomes. *Genome Res*, **22**(6), 1154–1162.

Gymrek, M., Willems, T., Guilmatre, A., Zeng, H., Markus, B., Georgiev, S., Daly, M. J., Price, A. L., Pritchard, J. K., Sharp, A. J., and Erlich, Y. (2016). Abundant contribution of short tandem repeats to gene expression variation in humans. *Nat Genet*, **48**(1), 22–29.

Highnam, G., Franck, C., Martin, A., Stephens, C., Puthige, A., and Mittelman, D. (2013). Accurate human microsatellite genotypes from high-throughput resequencing data using informed error profiles. *Nucleic Acids Res*, **41**(1), e32.

Hosmer Jr, D. W. and Lemeshow, S. (2004). *Applied logistic regression*. John Wiley & Sons.

Kong, A., Frigge, M. L., Masson, G., Besenbacher, S., Sulem, P., Magnusson, G., Gudjonsson, S. A., Sigurdsson, A., Jonasdottir, A., Jonasdottir, A., Wong, W. S. W., Sigurdsson, G., Walters, G. B., Steinberg, S., Helgason, H., Thorleifsson, G., Gudbjartsson, D. F., Helgason, A., Magnusson, O. T., Thorsteinsdottir, U., and Stefansson, K. (2012). Rate of de novo mutations and the importance of father's age to disease risk. *Nature*, **488**(7412), 471–475.

Li, H. and Durbin, R. (2009). Fast and accurate short read alignment with burrows-wheeler transform. *Bioinformatics*, **25**(14), 1754–1760.

McKenna, A., Hanna, M., Banks, E., Sivachenko, A., Cibulskis, K., Kernytsky, A., Garimella, K., Altshuler, D., Gabriel, S., Daly, M., and DePristo, M. A. (2010). The genome analysis toolkit: a mapreduce framework for analyzing next-generation dna sequencing data. *Genome Res*, **20**(9), 1297–1303.

Mirkin, S. M. (2007). Expandable dna repeats and human disease. *Nature*, **447**(7147), 932–940.

Sun, J. X., Helgason, A., Masson, G., Ebenesersdóttir, S. S., Li, H., Mallick, S., Gnerre, S., Patterson, N., Kong, A., Reich, D., and Stefansson, K. (2012). A direct characterization of human mutation based on microsatellites. *Nat Genet*, **44**(10), 1161–1165.

Veselinović, I. (2006). Microsatellite DNA analysis as a tool for forensic paternity testing (dna paternity testing). *Med Pregl*, **59**(5-6), 241–243.