Efficient and exact maximum likelihood quantisation of genomic features using dynamic programming

Mingzhou (Joe) Song*

Department of Computer Science,
New Mexico State University,
Las Cruces, NM 88003, USA
Fax: +1 575 646 1002
E-mail: joemsong@cs.nmsu.edu
*Corresponding author

Robert M. Haralick

PhD Program in Computer Science,
Graduate Center, City University of New York,
New York, NY 10016, USA
Fax: +1 212 817 1510
E-mail: haralick@ptah.gc.cuny.edu

Stéphane Boissinot

Department of Biology,
Queens College, City University of New York,
Flushing, NY 11367, USA
Fax: +1 718 997 3445
E-mail: stephane.boissinot@qc.cuny.edu

Abstract: An efficient and exact dynamic programming algorithm is introduced to quantise a continuous random variable into a discrete random variable that maximises the likelihood of the quantised probability distribution for the original continuous random variable. Quantisation is often useful before statistical analysis and modelling of large discrete network models from observations of multiple continuous random variables. The quantisation algorithm is applied to genomic features including the recombination rate distribution across the chromosomes and the non-coding transposable element LINE-1 in the human genome. The association pattern is studied between the recombination rate, obtained by quantisation at genomic locations around LINE-1 elements, and the length groups of LINE-1 elements, also obtained by quantisation on LINE-1 length. The exact and density-preserving quantisation approach provides an alternative superior to the inexact and distance-based univariate iterative k-means clustering algorithm for discretisation.

Keywords: quantisation; discretisation; dynamic programming; recombination rate distribution; transposable elements; LINE-1; retrotransposon.

Reference to this paper should be made as follows: Song, M., Haralick, R.M. and Boissinot, S. (2010) 'Efficient and exact maximum likelihood quantisation of genomic features using dynamic programming', *Int. J. Data Mining and Bioinformatics*, Vol. 4, No. 2, pp.123–141.

Biographical notes: Mingzhou (Joe) Song received his PhD in 2002 and MS in Electrical Engineering in 1999, both from the University of Washington, Seattle. He is an Assistant Professor of Computer Science, New Mexico State University since 2005. He was Assistant Professor of Computer Science with Queens College and Graduate Center, City University of New York from 2002 to 2005. His research areas include data mining, computational modelling, quantitative biology, and computer vision. He has collaborated with life scientists on computational modelling in biofuels, cancer, neuroscience, and microbiology.

Robert M. Haralick received his BA in Mathematics in 1964, BS in Electrical Engineering in 1966, MS in Electrical Engineering in 1967, and PhD in 1969, all from the University of Kansas. He is a Distinguished Professor of Computer Science, Graduate Center, City University of New York. He held the Clairmont Egtvedt Professorship in Electrical Engineering with University of Washington. He has made contributions in computer vision and pattern recognition, the most recent in manifold clustering in high dimensional spaces. He is a fellow of both IEEE and IAPR. He has served as editor for computer vision and pattern recognition journals.

Stéphane Boissinot received his PhD in 1994 from the University of Montpellier II, France. He was a Post-Doctoral fellow at the University of Texas, Houston from 1994 to 1996, and at the National Institutes of Health, Bethesda from 1997 to 2002. In 2003, he joined the Faculty in the Department of Biology at Queens College, the City University of New York, where he is currently Associate Professor. He is also Faculty at the Graduate Center of the City University of New York. His research interests are the evolution retrotransposons in vertebrate genomes and the evolution of resistance to viral infection.

1 Introduction

Quantisation is a monotonically increasing transformation that converts a continuous random variable to a discrete random variable. Quantisation functions that better preserve the original probability density function (p.d.f.) legitimise the transfer of statistical analysis and modelling performed on the discrete random variable back to the original continuous random variable. We present an efficient and exact algorithm that achieves such a density-preserving quantisation by dynamic programming. The optimality of the discretisation is guaranteed by a general mapped additivity satisfied by all major quantisation criteria. In our optimal quantisation algorithm, the most important regions are finely quantised, while less important regions are coarsely quantised, statistically much more efficient than a uniform quantisation. Other methods, e.g., kernel methods, treat everywhere in a space equally without the

prioritised resource allocation. For the less important regions, there is the potential wasting of resources. The algorithm can work on either continuous data points or counts of data already accumulated in finer-than-desired bins. The number of quantisation levels is determined by either the Bayesian information criterion – a function of the log likelihood, the sample size, and the number of quantisation levels, or cross validation.

Graphical modelling of multiple random variables has motivated continued research on quantisation algorithms. A graphical model uses a graph to represent the joint p.d.f. of multiple random variables. Each node in the graph represents a random variable. Edges between nodes encode statistical dependencies among variables. The joint p.d.f. can be decomposed to the product of conditional probability functions of variables at each node given their parent nodes. A graphical model of continuous random variables typically makes parametric assumptions on the conditional probabilities for each node in the graph, but not so for a graphical model of discrete random variables. Thus discretisation is often necessary for graphical modelling if no prior knowledge is available on the forms of conditional probabilities for each continuous random variable in question. Additionally, there are more alternatives (Margaritis and Thrun, 2001) to determine statistical independencies between discrete random variables than for continuous ones when the underlying p.d.f. is unknown.

Relevant to our work are approaches that find a quantisation of the data by optimising an objective function. Entropy (Haralick, 1976), likelihood (Hearne and Wegman, 1992), and distance have been used as objective functions. Among these criterions, only likelihood ties directly to the p.d.f. of the original continuous random variable. A less-known optimal solution (Wu, 1992) using dynamic programming has been provided for the univariate k-means problem. Fulton et al. (1995) have later used dynamic programming to find an optimal quantisation to classify a univariate sample. However, dynamic programming has not been used in density-preserving quantisation. Our methodology obtains a non-uniform quantisation by optimising an objective function that combines likelihood and entropy. Optimal quantisation ensures the adaptivity to the data and overcomes the statistical ineffectiveness of uniform quantisation.

We applied our quantisation algorithm to genomic features including the recombination rate and the distribution of Long Interspersed Nuclear Element LINE-1 (L1) in the human genome. The association pattern is studied between the recombination rate, obtained by quantisation at genomic locations around L1 elements, and the length groups of L1 elements, also obtained by quantisation on L1 length.

The paper is organised into seven sections. Following Section 1 the introduction, we define the density-preserving quantisation objective function in Section 2; the optimality condition for finding a quantisation by dynamic programming is discussed in Section 3; the dynamic programming algorithm for the quantisation is designed and analysed in Section 4; quantisation results of the recombination rate distribution function in human genome are presented in Section 5; the association of quantised length groups of L1 with the recombination rate is discovered in Section 6; finally, we draw our conclusions in Section 7.

2 The likelihood of quantisation

We define and justify a quantisation objective function that includes the likelihood and entropy measures on the observed data set. Let X be a continuous random variable with p.d.f. $p(x)$. Let calligraphic $\chi = \langle x_1, x_2, \ldots, x_N \rangle$ be a sorted sequence of a random sample of size N from X, where $x_1 \leq x_2 \leq \cdots \leq x_N$. We define χ_m^n as the subsequence $\langle x_m, x_{m+1}, \ldots, x_n \rangle$. Let Q be an L-level quantisation with decision boundaries $B = \{b_0, b_1, \ldots, b_L\}$, $b_0 < b_1 < \cdots < b_L$. Let $\Delta(q)$ be the width of bin q. Let N_q be the total number of data points in bin q. Let $\hat{p}(x)$ be the p.d.f. derived from the histogram of the observed data using quantisation Q.

To preserve the original p.d.f. $p(x)$, one can minimise the Kullback–Leibler divergence from $\hat{p}(x)$ to $p(x)$, defined as

$$D_{KL}(p \,\|\, \hat{p}) = \int p(x) \log \frac{p(x)}{\hat{p}(x)} \mathrm{d}x = \mathbf{E}[\log p(X)] - \mathbf{E}[\log \hat{p}(X)].$$

As $p(x)$ is fixed, minimising $D_{KL}(p \,\|\, \hat{p})$ is equivalent to maximising $\mathbf{E}[\log \hat{p}(X)]$. Let $\bar{p}_q$ be the estimated average probability density of bin q computed by

$$\bar{p}_q = \frac{N_q/N}{\Delta(q)}. \tag{1}$$

We estimate $\mathbf{E}[\log \hat{p}(X)]$ by the average sample log likelihood. Thus the *log likelihood* of X for quantisation Q is

$$J(X \mid Q) = \mathbf{E}[\log \hat{p}(X)] = \frac{1}{N} \sum_{q=1}^{L} N_q \log \bar{p}_q = \sum_{q=1}^{L} J(X \mid q), \tag{2}$$

where

$$J(X \mid q) = \frac{N_q}{N} \log \bar{p}_q$$

is the contribution from bin q.

While entropy has been utilised as a class impurity measure (Breiman et al., 1984), we use entropy to characterise the generalisation ability of quantisation. Maximising entropy corresponds to minimising information loss. Entropy is defined by

$$H(X \mid Q) = -\sum_{q=1}^{L} \frac{N_q}{N} \log \frac{N_q}{N} = \sum_{q=1}^{L} H(X \mid q), \tag{3}$$

where

$$H(X \mid q) = \frac{N_q}{N} \log \frac{N}{N_q}$$

is the contribution from bin q. Examples of maximum entropy quantisation include equal probability quantisation (Haralick et al., 1973), histogram equalisation

(Jain, 1989), Voronoi tessellation (Voronoi, 1908), or more generally, nearest neighbour partitions (Gersho and Gray, 1992).

In contrast to likelihood, entropy is not a direct performance measure of pattern recognition test results. Rather, the entropy measure in our context controls over-fitting. The larger the entropy, the less likely the possibility of over-quantisation.

We define the quantisation objective function or performance measure as

$$T(X \mid Q) = w_J \, J(X \mid Q) + w_H \, H(X \mid Q) \tag{4}$$

with

$$w_J + w_H = 1, \quad w_J, w_H \geq 0,$$

where w_J and w_H are given weights for log likelihood and entropy, respectively. The first term allows a best fit to the data while the second term prevents over-fitting. If we define $T(X \mid q)$, the contribution from bin q, as

$$T(X \mid q) = w_J \, J(X \mid q) + w_H \, H(X \mid q).$$

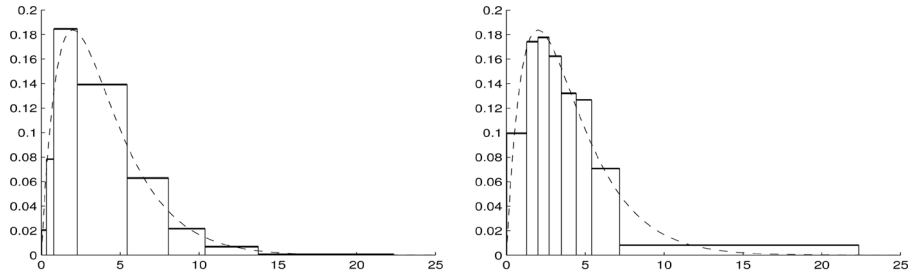
$T(X \mid Q)$ can be written in an additive form as

$$T(X \mid Q) = \sum_{q=1}^{L} T(X \mid q). \tag{5}$$

A data-driven strategy is to determine the coefficients w_J, w_H through cross validation. The values of w_J, w_H that maximise the likelihood of the left-out fold are selected to be the coefficients. The number of quantisation levels is determined by either the Bayesian information criterion – a function of the log likelihood, the sample size, and the number of quantisation levels, or cross validation.

Example: We illustrate with a Chi-squared example that contrasts maximum likelihood and maximum entropy quantisation. Our example has 1000 data points generated using a Chi-squared distribution with 4 degrees of freedom. The number of quantisation levels was 8. The density estimates are shown in Figure 1. The dashed line is the original Chi-squared p.d.f. In Figure 1(a), it is evident that the underlying density changes much more rapidly in $[0, 2]$ than in $[2, \infty)$. The bins are narrower for the region from 0 to 2 than for the region above 2, corroborating the consistency result in Scott (1992). In Figure 1(b), the bins for the region around the mode at 2 are narrower than the region further away from the mode. The density of the region around the mode is larger than other regions. When entropy is maximised, each bin contains about the same number of points. This naturally leads to narrower bins for regions of higher density and wider bins for regions of lower density. The rationale behind the entropy measure is that the least commitment should be made to the sample. This controls the generalisation ability of the quantisation. On the other hand, the maximum likelihood approach finds the best fit to the data and it may over-fit. Therefore, it is necessary to combine the two measures in a controlled fashion as we have done in defining $T(X \mid Q)$, which is especially important when the sample size is small.

Figure 1 Density estimates of Chi-squared data using optimal quantisation: (a) maximum likelihood quantisation ($w_J = 1, w_H = 0$) and (b) maximum entropy quantisation ($w_J = 0, w_H = 1$)



3 The optimality condition for quantisation using dynamic programming

Given the sorted data sequence χ and the number of quantisation levels L, the goal of quantisation is to find an optimal quantiser Q^* such that a pre-defined objective function $T(\chi \mid Q)$ is maximised by Q^*. An efficient solution of such a problem is still open for multivariate random variables. However, an efficient dynamic programming solution exists for optimal quantisation of a univariate random variable given that the quantisation performance measure satisfies a very general mapped additivity condition.

Definition 3.1 (Sub-quantiser)**:** Q_r^u is called a sub-quantiser of quantiser Q if it has $u - r + 1$ quantisation levels and the decision boundaries are the same with those for intervals from r to u of Q. We define $T(\chi_m^n \mid Q_r^u)$ as the performance measure of the sub-quantisation, evaluated on the subsequence χ_m^n of χ that falls in the bins of Q_r^u.

The performance measure of a sub-quantiser is exactly the contributions from the data points and intervals it covers. Notice that such defined sub-quantiser performance measure may be different from the performance measure of an isolated quantiser that covers just the same points and intervals. For the performance measure defined in equation (4) that involves equations (2) and (3), N is still the size of the overall data set χ_N even when computing sub-quantiser performance measures.

Definition 3.2 (Mapped additivity)**:** The mapped additivity condition is that the mapped performance measure of any quantiser Q on a given data set is additive over mapped performance measures of any combination of sub-quantisers of Q, when there is a monotonically increasing function that can achieve the mapping. Let $g(x)$ be such a monotonically increasing function defined on the domain of $T(\chi \mid Q)$. The mapped additivity can be written as

$$g(T(\chi \mid Q)) = \sum_{j=1}^{M} g\big(T\big(\chi_{m_j}^{n_j} \mid Q_{r_j}^{u_j}\big)\big), \quad \text{for any } Q, \quad 0 < M \leq L, \quad \text{and} \quad \chi. \tag{6}$$

Lemma 3.3 (Optimal sub-quantiser)**:** *Let quantiser Q^*, among all the quantisers that have L quantisation levels, maximise the mapped additive performance measure*

$T(\chi \mid Q)$ on the data set χ of size N. Let x_n be the largest element in interval q of quantiser Q^*. Then the sub-quantiser Q_1^{*q}, among all the sub-quantisers that have q quantisation levels and x_n as their largest element in interval q, maximises the performance measure $T(\chi_1^n \mid Q_1^q)$, i.e., $T(\chi_1^n \mid Q_1^{*q}) = \max_{Q_1^q} T(\chi_1^n \mid Q_1^q)$.

Proof by contradiction: By the mapped additive property of T,

$$g(T(\chi \mid Q^*)) = g\big(T\big(\chi_1^n \mid Q_1^{*q}\big)\big) + g\big(T\big(\chi_{n+1}^N \mid Q_{q+1}^{*L}\big)\big).$$

Since x_n is always the largest element of interval q, the second term $T(\chi_{n+1}^N \mid Q_{q+1}^{*L})$, which is the performance measure in the last $L-q$ intervals on data $\{x_{n+1}, \ldots, x_N\}$, would not be affected by the choice of Q_1^{*q}.

Assume that $\widehat{Q}_1^q$ was another sub-quantiser that quantises χ_1^n into q intervals with x_n being the largest element in interval q that does better in performance than Q_1^{*q}, that is,

$$T\big(\chi_1^n \mid \widehat{Q}_1^q\big) > T\big(\chi_1^n \mid Q_1^{*q}\big). \tag{7}$$

We could create a new quantiser $\widehat{Q}$ by combining the sub-quantiser $\widehat{Q}_1^q$ and Q_{q+1}^{*L}, which has the performance measure

$$\begin{aligned} g\big(T\big(\chi \mid \widehat{Q}\big)\big) &= g\big(T\big(\chi_1^n \mid \widehat{Q}_1^q\big)\big) + g\big(T\big(\chi_{n+1}^N \mid Q_{q+1}^{*L}\big)\big) \\ &> g\big(T\big(\chi_1^n \mid Q_1^{*q}\big)\big) + g\big(T\big(\chi_{n+1}^N \mid Q_{q+1}^{*L}\big)\big) \\ &= g\big(T\big(\chi \mid Q^*\big)\big). \end{aligned}$$

By the monotonically increasing property of $g(x)$, the above leads to

$$T(\chi \mid \widehat{Q}) > T(\chi \mid Q^*).$$

This conclusion contradicts the condition that $T(\chi \mid Q^*)$ is the maximum performance measure on χ_1^N among all quantisers with L levels. Then the assumption made in equation (7) must be incorrect. Thus

$$T\big(\chi_1^n \mid Q_1^{*q}\big) \geq T\big(\chi_1^n \mid \widehat{Q}_1^q\big) \tag{8}$$

must be true. Therefore, $T(\chi_1^n \mid Q_1^{*q})$ maximises the performance measure on the subsequence χ_1^n over q quantisation levels, that is,

$$T\big(\chi_1^n \mid Q_1^{*q}\big) = \max_{Q_1^q} T\big(\chi_1^n \mid Q_1^q\big). \qquad \square$$

Next, we establish the optimality of quantisation by dynamic programming under the mapped additivity condition.

Theorem 3.4: *If $T(\chi \mid Q)$ satisfies the mapped additivity condition defined in equation* (6)*, finding an optimal quantisation Q^* of L levels to maximise $T(\chi \mid Q)$ can be solved exactly using dynamic programming by the recurrence*

$$T[n,q] = \begin{cases} 0 & n = 0 \quad or \quad q = 0 \\ \max\limits_{1 \leq i \leq n} T[i-1, q-1] + g\big(T\big(\chi_i^n \mid Q_q^q\big)\big), & 1 \leq n \leq N, \quad 1 \leq q \leq L, \end{cases} \tag{9}$$

and the optimal performance measure is

$$T(\chi \mid Q^*) = \max_{Q} T(\chi \mid Q) = g^{-1}(T[N, L]).$$

Proof: By the recursive definition of $T[n, q]$ in equation (9), we must have

$$T[n, q] = \max_{Q_1^q} g\big(T\big(\chi_1^n \mid Q_1^q\big)\big),$$

due to Lemma 3.3, i.e., $T[n, q]$ must correspond to the optimal mapped performance measure that can be achieved for the first n points over q quantisation levels. Thus $T[N, L]$ corresponds to the optimal performance measure for the entire data set with L quantisation levels. Therefore, the inversely mapped value $g^{-1}(T[N, L])$ achieves the optimal performance measure $T(\chi \mid Q^*)$ obtained by an optimal quantiser Q^*. □

With $g(x) = x$ and under the constraint that a decision boundary in Q must be a middle point between some pair of consecutive distinct points, $T(X \mid Q)$ as shown in equation (5) meets the mapped additivity requirement. In addition to our definition of $T(X \mid Q)$, many problems in data mining involve performance measures that satisfy such a condition. Examples include k-means clustering operating in any metric space, and discretisation that maximises classification accuracy using either class purity entropy or percentage of correct classifications.

4 Maximum likelihood quantisation using dynamic programming

As the optimality condition equation (6) holds for $T(X \mid Q)$ when $g(x) = x$, we can use dynamic programming to find an optimal quantisation that maximises $T(X \mid Q)$. To avoid over-fitting, we require a minimum number of k data points in each bin and that identical ones are put into the same bin. We only set a decision boundary in the middle of two consecutive and distinct data points. This affects the range of $J(X \mid Q)$, but it is trivial when the sample size is not too small. This restriction prevents $J(X \mid Q)$ from overflow. Let T be an $(N + 1) \times (L + 1)$ matrix, whose entry $T[n, q](0 \le n \le N, 0 \le q \le L)$ is the maximum performance measure from bin 1 to q when x_n is the largest data in bin q. Let I be an $(N + 1) \times (L + 1)$ matrix, whose entry $I[n, q](0 \le n \le N, 0 \le q \le L)$ is the index to the smallest element in bin q such that $T[n, q]$ is achieved. Let T^1 be an $N \times N$ matrix, whose entry $T^1[i, n](1 \le i \le n \le N)$ is the performance measure contributed by a sub-quantiser with a single bin containing exactly x_i to x_n, that is,

$$T^1[i, n] = T\big(\chi_i^n \mid Q_q^q\big), \quad \forall q \in \{1, 2, \ldots, L\}.$$

The dynamic programming for finding a quantisation to maximise $T[N, L]$ is described below.

Initialisation – $T[n, q]$ is set to zero when either no point is covered $(n = 0)$ or no quantisation is applied $(q = 0)$ as in equation (10). $I[n, q]$ is initialised as in equation (11): $I[0, 0] = 0$ indicates the halting of backtrack; The -1 values indicate

that those locations are invalid as the quantisation would be on an empty set, have more levels than points, or have some empty bins.

$$T[n,q] = 0, \quad n = 0 \quad \text{or} \quad q = 0 \tag{10}$$

$$I[n,q] = \begin{cases} 0, & n = 0, \quad q = 0 \\ -1, & n = 0, \quad q > 0; \quad \text{or} \quad n > 0, \quad q = 0 \\ -1, & 0 \le q < \max(1, n - (N - L)), \quad n \neq 0, \quad q \neq 0 \\ -1, & \min(n, L) < q \le L, \quad n \neq 0, \quad q \neq 0 \end{cases}. \tag{11}$$

Feasible decision boundary index set: The indices of the feasible data for being the smallest element in bin q form the feasible decision boundary index set

$$\mathcal{A}_q^n = \{i \,|\, i \le n - k + 1, I[i-1, q-1] \neq -1, x_{i-1} \neq x_n, I[n,q] \neq -1, x_n \neq x_{n+1}\}.$$

The inequality $i \le n - k + 1$ guarantees that at least k data points are in bin q; $I[i-1, q-1] \neq -1$ states that x_{i-1} must be feasible for the largest point in the previous bin $q-1$; $x_{i-1} \neq x_n$ enforces that the feasible largest point in the previous bin $q-1$ must not be the same as x_n, to avoid splitting identical data points into different bins; $x_n \neq x_{n+1}$ is also not to split identical data points; $I[n,q] \neq -1$ asserts that x_n must be feasible for the largest point of bin q.

Recurrence: If $\mathcal{A}_q^n$ is empty, then $I[n,q] \triangleq -1$, meaning x_n does not qualify for the largest point in bin q. Otherwise,

$$T[n,q] \triangleq \max_{i \in \mathcal{A}_q^n} T[i-1, q-1] + T^1[i,n], \tag{12}$$

$$I[n,q] \triangleq \operatorname*{argmax}_{i \in \mathcal{A}_q^n} T[i-1, q-1] + T^1[i,n]. \tag{13}$$

Algorithm 1 fills matrices T and I row by row using the recurrence equations. The range limit of q in line 5 is equivalent to filling the lower left and upper left corners of matrix I with -1. The actual initialisation of the first column of I is implicit from lines 7 to 12. Line 15 decides the feasible decision boundary set. Lines 17 and 18 implement the recurrence equation if $\mathcal{A}$ is not empty. Matrix I is returned for backtracking.

Once matrix I is determined, an optimal quantisation can be retrieved by Algorithm 2. Backtracking starts from $I[N, L]$ and traces back to $I[0,0]$. Two dummy data points $-\infty$ and $+\infty$ are introduced in line 2. If a finite range quantiser is needed, we can set them to $x_1 - \delta$ and $x_N + \delta$ instead, where δ is a quantity not larger than the data resolution. When the performance measure contains the average log likelihood, we shall use finite width intervals. Since the value of $I[n,q]$ is the index to the smallest point in interval q if x_n is the largest point of interval q, $I[n,q] - 1$ must be the index to the largest point in interval $q-1$. The backtrack rewinds until $q = 0$. Each decision boundary is set to the middle of two adjacent points in different intervals (line 4).

Theorem 4.1: *The dynamic programming algorithm* (*Algorithm 1*) *has time complexity* $O(LN^2)$. *The backtrack algorithm* (*Algorithm 2*) *has time complexity* $O(L)$.

Algorithm 1 Find-Optimal-Quantisation($\mathcal{X}$, L, k)

Sort $\mathcal{X}$ in non-decreasing order if not already so
Initialise T and I;
for $n \leftarrow 1$ to N **do**
  Calculate the n-th column of T^1;
  for $q \leftarrow \max(1, n-(N-L))$ to $\min(n, L)$ **do**
    if $n \neq N$ and $x_n = x_{n+1}$ **then**
      $I[n,q] \leftarrow -1$;
    else if $q = 1$ **then**
      if $n \geq k$ **then**
        $T[n,q] \leftarrow T^1[1,n]$,$I[n,q] \leftarrow 1$;
      else
        $I[n,q] \leftarrow -1$;
      end if
    else
      $\mathcal{A} \leftarrow \{i | q \leq i \leq n-k+1,\ x_{i-1} \neq x_n,\ I[i-1,q-1] \neq -1, x_n \neq x_{n+1}\}$;
      if $\mathcal{A} \neq \emptyset$ **then**
        $T[n,q] \leftarrow \max_{i \in \mathcal{A}} T[i-1,q-1] + T^1[i,n]$;
        $I[n,q] \leftarrow \operatorname{argmax}_{i \in \mathcal{A}} T[i-1,q-1] + T^1[i,n]$;
      else
        $I[n,q] \leftarrow -1$;
      end if
    end if
  end for
end for
return I;

Algorithm 2 Backtrack($\mathcal{X}$, I)

$n \leftarrow N$, $q \leftarrow L$;
$x_0 \leftarrow -\infty$, $x_{N+1} \leftarrow +\infty$;
while $q \neq 0$ **do**
  $b_q \leftarrow \frac{x_n + x_{n+1}}{2}$;
  $n \leftarrow I[n,q] - 1$, $q \leftarrow q - 1$;
end while
$b_0 \leftarrow x_0$;
return Q;

Proof: $O(N \log N)$ is used in sorting the data. $O(LN^2)$ is used for filling in matrix T and I. Brute force calculation of matrix T^1 can take up to $O(N^3)$, immediately making the algorithm impractical to use when N is moderately large. Since $T^1[i,n]$ can be calculated from its neighbour $T^1[i-1,n]$ or $T^1[i+1,n]$ in constant time with minor memory costs, only $O(N^2)$ is used for filling in matrix T^1. So Algorithm 1 has $O(N \log N + N^2 + LN^2) = O(LN^2)$ as its overall time complexity.

$O(L)$ is spent backtracking the optimal intervals, since the while-loop has exactly L iterations and within each iteration it takes constant time. □

Theorem 4.2: *The dynamic programming algorithm* (*Algorithm 1*) *has space complexity* $O(LN)$.

Proof: In the most straightforward implementation, $N(N+1)/2$ would be needed to store matrix T^1, which can be reduced to linear space. When the nth rows of T and I are calculated, only the nth column of T^1 is used and this column will not be used again. Thus, during any iteration of the for-loop on n, we save only the nth column of T^1. This will reduce the space needed for T^1 from N^2 to N. We need $2LN$ space for matrices T and I. So the total space complexity is $O(2LN+N) = O(LN)$, which is the original claim. □

The dynamic programming algorithm taking sample points can be readily changed to apply to merge counts of data already accumulated in finer-than-desired bins, because the performance measure uses only counts of data within a bin and the bin widths rather than the actual values of those points.

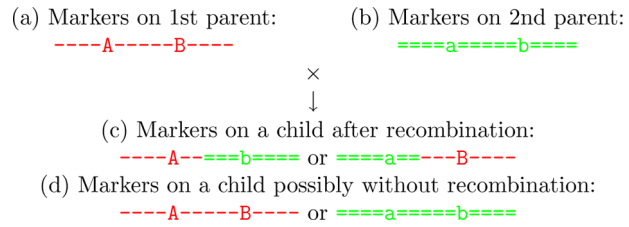
5 Estimation of recombination rate distribution over chromosomes by quantisation

Recombination is a biological phenomenon that is of central importance to the fields of genetics and evolutionary biology. In the nucleus of each human cell (except the haploid gametes) each chromosome (except the sex chromosomes) comes in two copies called homologous chromosomes, one chromosome coming from the mother and one from the father. During meiosis (that is the formation of four haploid gametes from a diploid cell) homologous chromosomes exchange their genetic materials in a process called recombination. Thus, the chromosomes at the next generation do not contain the same genetic information as the parent's chromosomes but instead are a mosaic of alleles from the mother's and father's chromosomes. The study of recombination is important to molecular evolution because the local rate of recombination affects the efficiency of natural selection. *Recombination Rate* (RR) is defined as the number of recombination events in a unit length of chromosome in terms of base pairs (bps), usually in centiMorgan per Mbps (cM/Mb). The RR Distribution (RRD) function maps a location on the chromosome to an RR value. However, observing recombination events has been limited due to the cost of experiments. As the complete human genome physical map becomes available, an accurate quantitative representation of the RRD becomes possible.

Recombination events are identified using both genetic and physical maps. On a genetic map, each marker represents a unique feature. A marker has two or multiple forms, called *alleles*. The alleles can be identified by polymerase chain reactions. Locations of markers on the physical map are determined in advance. Markers make detection of recombination events possible without sequencing the entire genomes of generations. The resolution of the identified events increases with the number of markers. This method is illustrated in Figure 2. The first parent has 2 markers A and B (Figure 2(a)) and the second parent has the same markers but with different alleles a and b (Figure 2(b)). If a child has the markers as in Figure 2(c), then at least one recombination event has occurred at some location between markers A and B. If a child has the same alleles as their parents as in Figure 2(d), then it is unlikely

to have a recombination event between A and B if the markers are close enough. This method cannot detect the exact location of a recombination event. It may miss a recombination event between markers. In addition, if the two parents carry the same set of alleles, no recombination event between the markers may be identified. Therefore, selection of markers directly affects the effectiveness of recombination detection. Typically, a good marker collection should be abundant, hyper-variable, and evenly distributed across the genome. One such family of markers is microsatellites, which are short sequences of motifs in tandem (Brown, 1999). The motifs can be di-, tri-, or tetra-nucleotide repeat units. In the Marshfield recombination map (Yu et al., 2001), over 8,000 microsatellites are used; in the Iceland recombination map (Kong et al., 2002), there are 5,000 microsatellites.

Figure 2 Identifying a recombination event with markers. One marker has two alleles A and a; the other has two alleles B and b (see online version for colours)



The frequency of recombination is not uniform across the genome: more frequent near the *telomere* – the end of a eukaryotic chromosome – and less frequent at the *centromere* where two copies of the homologous chromosomes hold together. We consider X, the location of a recombination event, a random variable. Let $p(x)$ be its p.d.f. Let $F(x)$ be its cumulative distribution function (c.d.f.).

The RRD function $R(x)$ is in proportion to $p(x)$ defined as $R(x) = R_0 p(x)$, where R_0 is the total amount of recombination events observed on a single chromosome of an individual. This definition is used in the Iceland RRD estimation (Kong et al., 2002). Since its exact physical location is unknown, a recombination event between two markers is assigned the position of the marker with larger coordinate on the chromosome. With N recombination event locations $x_1, x_2, \ldots, x_N$ observed, an estimated p.d.f. $\hat{p}(x)$ is obtained using the Parzen window method in Kong et al. (2002)

$$\hat{p}(x) = \frac{1}{N} \sum_{i=1}^{N} k(x, x_i), \tag{14}$$

where

$$k(x, x_i) = \begin{cases} \dfrac{1}{\Delta}, & |x - x_i| \leq \dfrac{\Delta}{2} \\ 0, & \text{otherwise} \end{cases},$$

and Δ is the bandwidth. Then they choose a sequence of M equally spaced locations $y_0, 2y_0, 3y_0, \ldots, My_0$ to calculate the estimated p.d.f. values. In the end, they fit splines to these points to obtain a smooth p.d.f $p(x)$ and then obtain $R(x)$. The critical bandwidth parameter Δ is 3 Mbps. The sample is drawn from 1257 meioses.

Another RRD is defined by $R(x) = R_0 \frac{\mathrm{d}F(x)}{\mathrm{d}x}$, used by the Marshfield RRD (Yu et al., 2001). In this approach, it is not necessary to know the exact location of each recombination event. They compute the empirical c.d.f. $\widehat{F}(x)$ from the observed recombination events, fit cubic splines to $\widehat{F}(x)$, and then obtain the RRD. In this study, only 184 meioses are analysed to identify recombination events, which is a much smaller sample size compared to Kong et al. (2002).

The RRDs in Kong et al. (2002) are represented as continuous functions, with empirically chosen bandwidth Δ. All the splines are saved and must be evaluated to calculate RRD at a location.

Alternatively, we performed optimal quantisation on the genetic distances of selected markers (Kong et al., 2002), given as the empirical c.d.f. of the recombination events. We first obtained the control parameters w_J, w_H, L, and k by a 5-fold cross-validation. The values of w_J and w_H range from 0 to 1 with a step of 0.1. L ranges from 2 to 2^8 in powers of 2. k ranges from 1 to 3^6 in powers of 3. Second, using the best parameters, a p.d.f. was estimated, on all the recombination events for each chromosome. The estimated RRD functions of chromosomes 3 and X are shown in Figures 3 and 4. Recombination is much more active around the ends of chromosomes than the centres. Our RRDs show more fluctuations than those shown in Kong et al. (2002), Yu et al. (2001). Since our control parameters are all cross-validated, it is very likely that the RRDs indeed change more abruptly than the much more smooth curves published before. To fit splines on our estimation result could make the curve smoother, but it requires validation of the smoothness. We further compare quantitatively the performance of optimal quantisation with the Parzen window method. To make the comparison fair, we did not apply splines. The evaluation is done by a 5-fold cross-validation. The performance measure is the log likelihood of the left-out data reserved for test, using the p.d.f. estimated from the data not using the left-out data. The average and the standard deviation of the cross-validated log likelihood for each chromosome are shown in Table 1. The average log likelihoods of the p.d.f. obtained by optimal quantisation are consistently higher than those by the Parzen window method. The standard deviations of both are similar, with Parzen window results slightly smaller on most of the chromosomes. Therefore the optimisation quantisation approach provides a better RRD estimation than that of the Parzen window.

Figure 3 Chromosome 3 (see online version for colours)

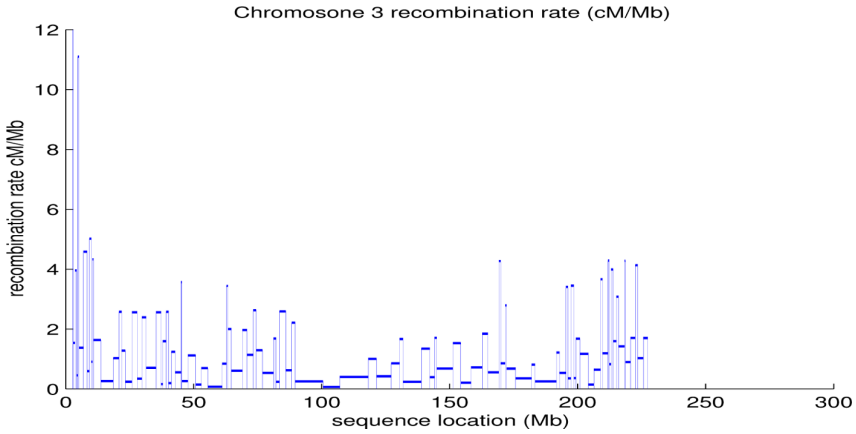


Figure 4 Chromosome X (see online version for colours)

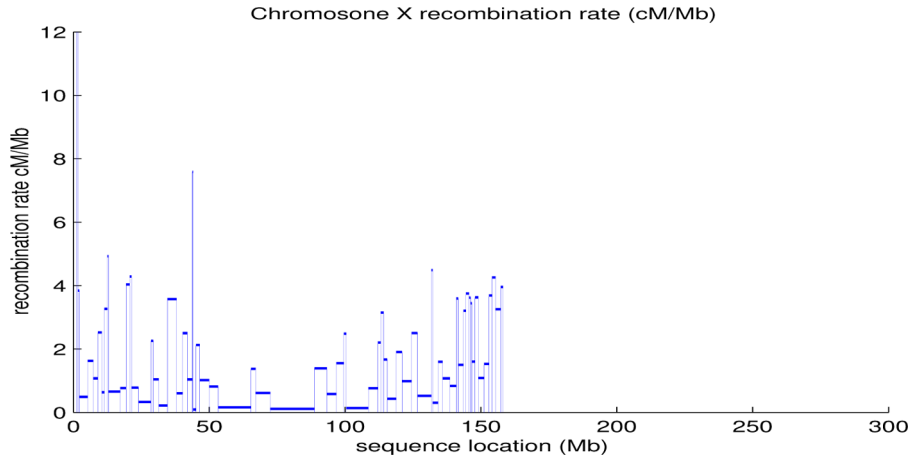


Table 1 Comparison between optimal quantisation and Parzen window

	Average log likelihood		*Standard deviation*	
Chromosome	*Quantisation*	*Parzen window*	*Quantisation*	*Parzen window*
1	−19.11	−19.17	0.03	0.01
2	−19.10	−19.21	0.05	0.02
3	−18.90	−19.05	0.04	0.04
4	−18.91	−18.98	0.03	0.02
5	−18.79	−18.91	0.04	0.03
6	−18.72	−18.88	0.05	0.03
7	−18.69	−18.87	0.03	0.02
8	−18.60	−18.78	0.02	0.01
9	−18.42	−18.52	0.04	0.03
10	−18.55	−18.69	0.05	0.05
11	−18.53	−18.65	0.06	0.03
12	−18.57	−18.63	0.03	0.04
13	−18.02	−18.32	0.06	0.04
14	−17.94	−18.14	0.07	0.07
15	−17.87	−18.17	0.06	0.07
16	−18.05	−18.18	0.07	0.04
17	−17.99	−18.14	0.05	0.05
18	−18.04	−18.16	0.08	0.06
19	−17.70	−17.95	0.09	0.05
20	−17.62	−17.70	0.09	0.03
21	−17.05	−17.28	0.06	0.05
22	−16.96	−17.16	0.08	0.05
X	−18.42	−18.53	0.04	0.03

6 Localised study of recombination rate within length groups of L1s

L1 retrotransposons have significantly affected the structure and function of mammalian genomes, including the human genomes. They have been a source of

genetic novelty and their activity accounts for at least 30% of the size of our genome. However, their replicative success is difficult to reconcile with the potential damages they can impose on their host's genome. The effect that L1 elements can have on the fitness of individuals remains a matter of debate. One approach used to understand their impact is to look at their distribution in the genome relative to the local recombination. The rationale is that if L1 elements of a given length are deleterious they should accumulate in regions of low recombination.

Therefore we decided to examine how RR near an L1 element depends upon the length of the element. A linear regression could not adequately capture subtlety of the RR-length interaction. Given the relatively large sample size of L1s, instead of fitting a higher order linear regression model, we analyse families of different age separately using the classification of Khan et al. (2006). We studied five families, named L1PA2 to L1PA6, and broke elements within each family into groups based on the length of the elements. We then looked at the trend of RR within each group. Grouping is determined by optimal quantisation of the lengths of all L1s under consideration. Intuitively, this method separates L1s into groups by length when there is a sudden change in the number of L1s over unit length. We selected the number of groups to be six, roughly capturing the overall distribution of length while assuring that the intervals are not too small for a meaningful regression. The six length groups are shown in Table 2. The grouping reflects a natural tendency for L1 to segregate by length.

Table 2 L1 groups by length, with length ranges, counts, and percentage

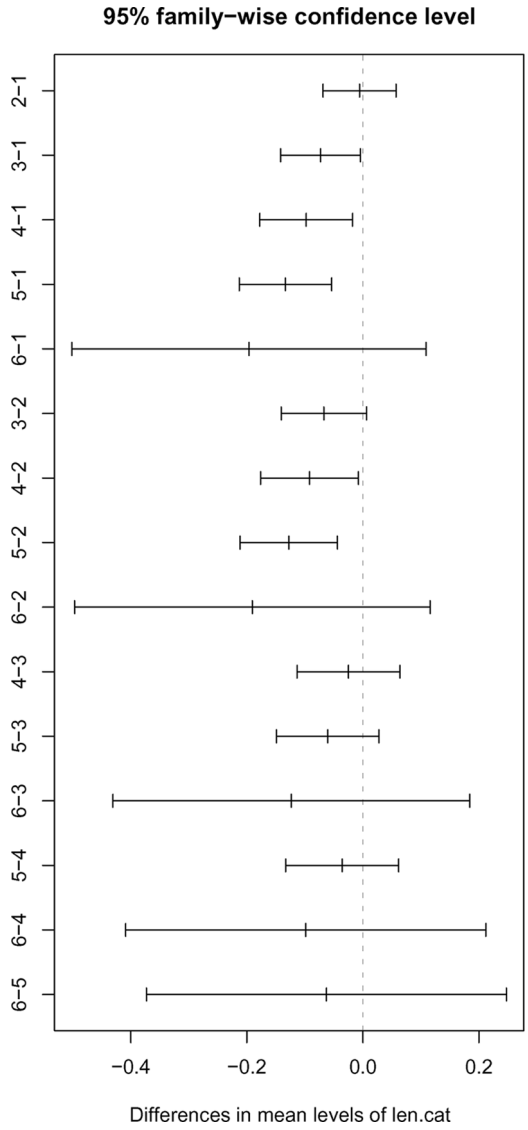
L1 groups	*Length range*	*L1 count/percentage* (%)
1	[100, 490]	12,226/34
2	[491, 1152]	8559/24
3	[1153, 2498]	6462/18
4	[2499, 6001]	4182/12
5	[6002, 6183]	4231/12
6	$\geq$6184	218/1

A one-way ANOVA (Table 3) indicates indeed the RR means are significantly different among L1 length groups. The Tukey's Honest Significant Differences (HSD) test reveals further details in Figure 5. Under the null hypothesis of RR mean equality across groups, if one compares every two groups using the 5% α-level, the chance of observing some inequality among the pairs can be much greater than the anticipated 5% type I error. The Tukey's HSD test corrects this problem. In Figure 5, the range of each line segment manifests the 95% confidence interval of the mean RR difference between the two length groups labelled on the left of the segment. The vertical dashed line marks the zero difference location. If an interval contains zero, there is no significant evidence from the sample to conclude that the two groups have different mean RRs. All differences are the mean RR of a group with a longer length minus that of one with a shorter length. A major observation is that no segments have both ends above zero, suggesting no significant trend of increasing RR as length increases. The only almost significant negative difference between two consecutive length groups occurs from groups 2 to 3, which accounts for other significant differences among non-consecutive length groups. Therefore, the multiple comparison analysis pins down that the most significant reduction in RR takes place among the L1s of intermediate length, that is between elements shorter and longer than 1.2 Kb.

Table 3 One-way ANOVA for RR over the length groups

	Degrees of freedom	*Sum of squares*	*Mean squares*	*F value*	$Pr(>F)$
Group	5	93	19	7.5441	4.330e-07
Residuals	35,872	88,107	2		

Figure 5 Tukey's HSD test on the RR means among length groups. Numbers on the vertical axes correspond to length groups. For example, 5–3 stands for the mean RR of group 5 minus that of group 3

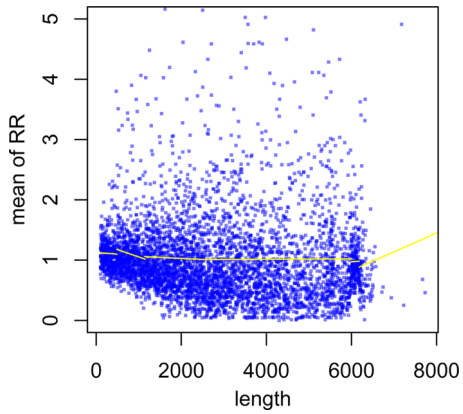


Based on the Tukey's HSD results, we studied the trend of RR within each length group using linear regression on the length of L1. The intercepts and slopes of each linear regression line, and the corresponding p-values are given in Table 4. No length group shows a significant positive slope. We observe that length group 2 has a highly significant negative slope. Figure 6 shows the mean RR-length scatter plot with the regression lines overlaid. We can observe in the plot a decreasing trend of the regression line in group 2 quite evidently. It is also quite evident subjectively that there is a declining tendency in the mean RR as the length increases. This further analysis match well to previous findings by the Tukey's HSD test. Therefore the major RR reduction occurs on the L1s of length 491 to 1152, which are not full-length L1s, but L1s of intermediate length.

Table 4 Linear regression slopes of each group

	Estimate	*Std. error*	*t-Statistic*	$Pr(>\|t\|)$
1:length	−5.537e-05	1.275e-04	−0.434	0.6641
2:length	−2.446e-04	9.006e-05	−2.716	0.0066
3:length	−3.409e-05	5.126e-05	−0.665	0.5060
4:length	3.042e-06	2.268e-05	0.134	0.8933
5:length	5.923e-05	4.409e-04	0.134	0.8931
6:length	3.108e-04	5.386e-04	0.577	0.5639

Figure 6 Scatter plot of mean RR vs. L1 length. The line segments are linear regressions within each group. Only the second segment has a significant decreasing trend (see online version for colours)



7 Conclusion

We have described a dynamic programming algorithm to quantise a random variable to preserve maximally the p.d.f. of the original continuous variable. Although our algorithm has a quadratic running time in sample size, it guarantees the optimality of quantisation. The distance-based k-means algorithm for univariate quantisation, popular simply due to its computational convenience, shall either be replaced by our

maximum likelihood approach when preservation of the distribution of the original continuous random variable is desired, or by a dynamic programming implementation similar to ours that guarantees optimality. Applications of our algorithm in estimating RR distributions and characterising L1 elements show its effectiveness in capturing the underlying p.d.f.s of data. It can also be used to discretise other genomic features including GC-content, gene expression rate, and non-coding element densities over a genome.

Acknowledgements

The authors thank the support from grants made by PSC-CUNY, CUNY Institute for Software Design and Development, and NSF CREST Center for Excellence in Computational Biology and Bioinformatics (Grant Number: HRD_0420407).

References

Breiman, L., Friedman, J.H., Olshen, R.A. and Stone, C.J. (1984) *Classification and Regression Trees*, Wadsworth & Brooks/Cole, Pacific Grove, California.

Brown, T.A. (1999) *Genomes*, Wiley-Liss, New York.

Fulton, T., Kasif, S. and Salzberg, S.L. (1995) 'Efficient algorithms for finding multi-way splits for decision trees', *Proc. 12th Int'l Conf. on Machine Learning*, Morgan Kaufmann, Tahoe City, California, USA, pp.244–251.

Gersho, A. and Gray, R.M. (1992) *Vector Quantization and Signal Compression*, Kluwer Academic Publishers, Boston.

Haralick, R.M. (1976) 'The table look-up rule', *Communications in Statistics – Theory and Methods*, Vol. A5, No. 12, pp.1163–1191.

Haralick, R.M., Shanmugam, K. and Dinstein, I. (1973) 'Textural features for image classification', *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-3, No. 6, pp.610–621. See Appendix for equal-probability quantization.

Hearne, L.B. and Wegman, E.J. (1992) 'Maximum entropy density estimation using random tessellations', *Computing Science and Statistics*, Vol. 24, pp.483–487.

Jain, A.K. (1989) *Fundamentals of Digital Image Processing*, Prentice-Hall, Englewood Cliff, NJ.

Khan, H., Smit, A. and Boissinot, S. (2006) 'Molecular evolution and tempo of amplification of human LINE-1 retrotransposons since the origin of primates', *Genome Research*, Vol. 16, pp.78–87.

Kong, A., Gudbjartsson, D.F., Sainz, J., Jonsdottir, G.M., Gudjonsson, S.A., Richardsson, B., Sigurdardottir, S., Barnard, J., Hallbeck, B., Masson, G., Shlien, A., Palsson, S.T., Frigge, M.L., Thorgeirsson, T.E., Gulcher, J.R. and Stefansson, K. (2002) 'A high-resolution recombination map of the human genome', *Nature Genetics*, Vol. 31, pp.241–247.

Margaritis, D. and Thrun, S. (2001) 'A Bayesian multiresolution independence test for continuous variables', *17th Conference on Uncertainty in Artificial Intelligence (UAI)*, Seattle, Washington, pp.346–353.

Scott, D.W. (1992) *Multivariate Density Estimation – Theory, Practice and Visualization*, John Wiley & Sons, New York.

Voronoi, G. (1908) 'Nouvelles applications des parametres continus a la théorie des formes quadratiques, deuxieme memoire, recherches sur les parallelloedres primitifs', *Journal für die Reine und Angewandte Mathematik*, Vol. 134, pp.198–287.

Wu, X. (1992) 'Color quantization by dynamic programming and principal analysis', *ACM Trans. Graph.*, Vol. 11, No. 4, pp.348–372.

Yu, A., Zhao, C., Fan, Y., Jang, W., Mungall, A.J., Deloukas, P., Olsen, A., Doggett, N.A., Ghebranious, N., Broman, K.W. and Weber, J.L. (2001) 'Comparison of human genetic and sequence-based physical maps', *Nature*, Vol. 409, pp.951–953.