Supplemental Material

Crunch: Integrated processing and modeling of ChIP-seq data in terms of regulatory motifs

Severin Berger[1,2], Mikhail Pachkov[1], Phil Arnold[1,3], Saeed Omidi[1,4], Nicholas Kelley[1,3], Silvia Salatino[1,5] & Erik van Nimwegen[1,*]

[1]Biozentrum, University of Basel, and Swiss Institute of Bioinformatics, Basel, Switzerland
[2]Champalimaud Centre for the Unknown, Lisbon, Portugal
[3]Novartis Institutes for Biomedical Research, Basel, Switzerland
[4]Sophia Genetics, St-Sulpice, Switzerland
[5]Wellcome Centre for Human Genetics, University of Oxford, Oxford, UK
[*]E-mail: erik.vannimwegen@unibas.ch

Contents

1 Supplemental Methods

1.1 Quality Control and Adapter Removal

Raw sequencing reads are typically provided in FASTQ format and Crunch directly takes such FASTQ files as input. These raw reads can have diverse quality due to uncertainty in base calling and errors in the sequencing itself, and they potentially contain artefacts such as sequenced parts of sequencing adapters. To avoid contaminating downstream analyses with low quality or erroneous sequences, we perform an initial quality filtering, followed by adapter removal, and a final round of quality filtering as follows.

In the first round of quality filtering we discard reads that are either shorter than 25 nucleotides (nt), contain more than 2 ambiguous nucleotides (N's), or have an average Phred base calling quality score below 20 (corresponding to an error rate of 1%). As sequencing quality tends to decrease from the 5' to the 3' end of the read, we select the longest 5' prefix of the read that has an average Phred score of at least 20. The chosen prefix is maintained if it has a minimum length of 25 nt.

In the adapter removal step, we focus exclusively on 3' adapters, i.e. adapters that get (partially) sequenced if the sequence of interest (i.e. the fragment) is shorter than the length of the sequenced read. Crunch first aims to determine, for any given data set, which 3' adapter was used. Crunch uses a list of known Illumina adapters [1], which can be extended with custom adapters by the user if desired. For each adapter in the list, prefixes of lengths 14, 16, 18 and 20 nt are mapped to 250'000 randomly chosen reads from the data set, allowing up to 2 mismatches. The adapter with the highest number of matches is chosen as the putative adapter sequence. This adapter sequence is then trimmed stringently from the reads according to the procedure previously described in [2]. In particular, we trim adapters from reads using the following three steps: First, reads are scanned for full length adapter sequence matches allowing for 2 mismatches. All matched reads then get truncated starting at the beginning of the adapter sequence read match. Second, for the remaining reads, adapter sequence prefixes get matched to read suffixes allowing for 1 mismatch for matches longer than 6 nt and 2 mismatches for matches longer than 9 nt (i.e. as described previously in [2]). All matched read suffixes are then removed.

After adapter removal, we first remove all sequences whose remaining length is less than 14 and additionally remove low complexity reads. We define the complexity of a read as the entropy $H = -\sum_{\alpha,\beta} f_{\alpha\beta} \log(f_{\alpha\beta})$, with $f_{\alpha\beta}$ the frequency of dinucleotide $\alpha\beta$ in the read. All reads with $H < 0.5\log(16)$ are removed [2].

1.2 Mapping

Crunch maps the remaining reads to the reference genome using Bowtie version 1.1.1 [3]. Bowtie's parameters are set such that for every read all mapping positions with the least number of mismatches get reported (-a –strata –best) allowing for at most three mismatches (-v 3) and skipping reads when the number of mapping positions exceeds 100 (-m 100). Multi-mapping reads are uniformly distributed over all mapping positions, i.e. each mapping of a read is assigned a weight equal to one over the number of mapping positions. We store these aligned reads in a BED-like format called BEDWEIGHT, which besides specifying the location of the mapping, also specifies the mapping's weight. To allow visualization of the ChIP profiles in a genome browser, Crunch produces downloadable WIG files of the aligned reads.

1.3 Fragment Size Estimation

Crunch estimates the fragment size by finding the distance d that maximizes the correlation function

$$C(d) = \sum_{i \notin R} r_{+}(i) r_{-}(i+d) \tag{1}$$

where $r_+(n)$ and $r_-(n)$ are indicator functions that equal one if a read occurs at position n of the, respectively, plus or minus strand, and zero if no read occurs. The sum in equation (1) is over all genomic positions excluding regions annotated as repeats (which we denoted by R). Note that we use indicator functions rather than raw read counts to avoid the correlation function $C(d)$ to be dominated by a few positions with large read counts. $C(d)$ is computed for d ranging from 0 to 600, and typical resulting cross-correlation functions are depicted in Suppl. Fig. S3.

As has been observed previously, e.g. [4], for some datasets a local maximum in $C(d)$ occurs at d equal to the read length. Although repeat regions are masked to compute $C(d)$ we believe that this is an artefact deriving from reads mapping to repeats, which is supported by the fact that this peak in $C(d)$ becomes much more pronounced when repetitive regions are included. In particular, there are likely differences between the repetitive regions in the genome from which the data derives and the reference assembly to which the reads are mapped. For reads that derive from repetitive regions that are not represented in the reference genome, a fraction will likely not be able to map at all, but another fraction will by chance be able to map to a similar repeat in the reference genome, that locally happens to differ by 3 or less mismatches. These isolated matches will cause an over-representation of read pairs that are precisely one read length apart. To avoid such artefactual fragment length estimates, we only consider local optima in $C(d)$ for d larger than the read length.

1.4 Fragment counts in sliding windows

The first step in our peak calling procedure is to identify genomic regions that are enriched for fragments from the chromatin immunoprecipitation. Conceptually, if there is no binding of the immunoprecipitated protein at the locus, then the difference in ChIP and background fragment densities should result from random fluctuations, while when there is binding at the locus, then the ChIP fragment density will be higher than in the background sample. To this end we will have to compare, genome-wide, the fragment densities in the ChIP and background samples.

First, for each mapped read in both ChIP and background sample, we estimate the central position of the corresponding fragment to be half a fragment size toward the 3' direction, i.e. forwards for reads on the plus strand and backwards for reads on the minus strand. Next, we count the number of ChIP fragments in sliding windows along the genome. By default Crunch uses sliding windows of length 500, shifting windows by 250 bps at a time. The choice of window length 500 is a tradeoff between obtaining sufficient mapped reads to measure local fragment density with reasonable accuracy, and obtaining sufficient spatial resolution to ensure that windows cover only one or a few binding peaks (a single binding peak's width is roughly twice the fragment length, see below). If desired, the user can change the window length and the size of the shift between neighboring windows.

For each ChIP window, we count the number of reads in the background sample in a window of length 2000 (by default) centered on the same position as the ChIP window. Since the background fragments are approximately uniformly distributed, the background density fluctuates more slowly than for the ChIP sample (which has sudden peaks) and the average fragment density is also much lower than in ChIP binding peaks, so that a larger window is required to accurately estimate the background density. The user can again change this default if desired. The result is two vectors of fragment counts n and m for the ChIP and corresponding background windows, where fragment counts of possible ChIP or background replicates are summed.

1.5 Identification and removal of windows with high read densities in the background samples

We developed the following procedure to filter out regions with unusually high background signal. Reasoning that the distribution of fragment counts for normal background signals should have a roughly exponential tail, we fit the tail of the reverse cumulative distribution of the background fragment counts to an exponential distribution and determine the point at which the observed distribution starts deviating more than $\exp(0.5)$ vertically from the fitted exponential tail (this point corresponds to the red line in Fig. 2A of the main text). All windows with counts above this cut-off are excluded from further analysis.

1.6 Fitting the mixture model for enriched and unenriched regions

By taking the derivative of the log-likelihood $L = \sum_i \log[P_{mix}(n_i|N, m_i, M, \sigma, \mu, \rho)]$, where P_{mix} is defined in the Methods section of the main text, with respect to our parameters μ and ρ and setting it equal to 0, we get the following implicit equations that we solve by expectation maximization:

$$\mu = \frac{\sum_i p_{bg,i} \frac{\log(n_i/N) - \log(m_i/M)}{2\sigma^2 + 1/n_i + 1/m_i}}{\sum_i \frac{p_{bg,i}}{2\sigma^2 + 1/n_i + 1/m_i}}$$

and

$$\rho = \frac{\sum_i p_{bg,i}}{T},$$

with

$$p_{bg,i} = \frac{\rho P(n_i, m_i|\sigma, \mu, \rho, N, M)}{\rho P(n_i, m_i|\sigma, \mu, \rho, N, M) + (1-\rho)/W},$$

and T the number of windows in the genome.

To fit σ^2 we use a binary search to find the optimum of the likelihood with respect to σ^2. The derivative of the log-likelihood function with respect to $s = \sigma^2$ is given by

$$\frac{\partial L}{\partial s} = \sum_i p_{bg,i} \Big[\frac{(\log(n_i/N) - \log(m_i/M) - \mu)^2}{2(2\sigma^2 + 1/n_i + 1/m_i)^2} - \frac{1}{2(2\sigma^2 + 1/n_i + 1/m_i)} \Big]$$

To perform the binary search one first needs to bracket the zero of this equation. As the derivative $\partial L/\partial s$ is guaranteed to be negative as σ^2 goes to infinity, we first determine an upper bound of the search space by finding a value for σ^2 that yields a negative derivative. We use 0.001 as a lower bound for σ^2, as it is unrealistic that the multiplicative noise can be lower than this in practice. We then find the value of for which $\partial L/\partial s = 0$ by a binary search in this interval.

1.7 False Discovery Rate cut-off

All windows are sorted by their z-scores (started from the highest) and for each window w, the posterior probability is calculated that the window is a false positive, i.e. derives from the background distribution of unbound regions:

$$P_{\text{FP}}(w|D, \sigma, \mu, \rho) = \frac{\rho P_u(n_w|N, m_w, M, \sigma, \mu)}{\rho P_u(n_w|N, m_w, M, \sigma, \mu) + (1-\rho) P_b(n_w|N, m_w, M)},$$

where ρ is the overall fraction of background windows, n_w is the foreground count in the window, N the total foreground count in the genome, m_w the background count in the window, M the total background count, μ and σ are the fitted parameters of the background distribution, P_u is the probability of the read count n_w assuming the region is unbound, and P_b is the probability of the read count n_w assuming the window is bound (see the Methods section of the main text for details).

If we select the first T windows in our list, the estimated overall false discovery rate is given by:

$$\text{FDR}_T = \frac{1}{T} \sum_{i=1}^{T} P_{\text{FP}}(w_i|D, \sigma, \mu, \rho)$$

We then set T to the maximum for which $\mathrm{FDR}_T \leq 0.1$. The z-score of the Tth window then defines the z-score threshold z_*.

1.8 Widths of individual binding peaks

Theoretically, if all fragments were exactly the same length, if the breaking of the DNA during library preparation were completely random, and if the probability to immunoprecipitate a fragment were independent of where in the fragment the protein was bound, then each single binding site would result in an isosceles triangular coverage distribution with a base of two times the fragment size, centered on the actual binding site. However, fragment sizes will fluctuate, the propensity for the DNA to break will vary along the genome, and the probability to immunoprecipitate a fragment may depend on the relative position at which the protein is bound. Emprically, we typically observe approximately Gaussian shaped distributions of read coverage (e.g. Fig. 2D of the main text).

As discussed above, since the width of individual binding peaks only depends on the typical fragment size, all individual binding peaks in the coverage profiles should have similar widths. By constraining the widths of the fitted Gaussians in the second step of peak calling, we can therefore help the mixture model to detect true binding peaks. For this we examined the widths, i.e. the σ's of the Gaussian shaped distributions in all significantly enriched regions from the first step of peak calling from all our 123 ENCODE ChIP-seq data sets (see Results section) where fragment sizes range from 82 to 198 nucleotides. We observed that peak widths indeed scale linearly with fragment size (Suppl. Fig. S14).

By performing simple least squares linear regression on these data we determined that the average peak width equals $0.416 *$ fragmentsize $+ 17.1$ with a standard deviation of the residuals of 19. Using this result, we conservatively constrain peak widths by adding and subtracting $1.5 * 19$ to the average peak width to set the maximal and minimal peak widths, respectively. With these bounds, more than 90% of the peak widths from the 123 experiments are captured. We note that the peak width scaling that we observe is according to basic theoretical expectations. Assuming that the hypothesized isosceles triangular distribution and the observed Gaussian distribution share the same width at half height, we expect the σ of a Gaussian to scale as $(2\sqrt{2\log(2)})^{-1} *$ fragmentsize, i.e. $0.425 *$ fragmentsize, which is very close to our fit above. We hypothesize that the nonzero vertical offset might result from the fact that the binding of proteins to the binding site may suppress DNA breakage at these points, thereby slightly increasing the fragment size around binding sites relative to the average fragment size genome-wide.

1.9 Fitting a Gaussian mixture model to the coverage profile of each enriched region

To decompose each enriched region into individual binding events we approximate the data as if the coverage $C(i)$ at each position i in the region were an independent observation, the likelihood of the mixture model takes on the following form:

$$L(\vec{C} \mid \vec{\mu}, \vec{\sigma}, \vec{\rho}) = \prod_{i=1}^{l} \left[\sum_j \rho_j \frac{1}{\sqrt{2\pi\sigma_j^2}} \exp\left(- \frac{(i-\mu_j)^2}{2\sigma_j^2} \right) + \left(1 - \sum_j \rho_j \right) \frac{1}{l} \right]^{C_i}, \qquad (2)$$

where i runs over all positions from 1 to l in the region, $C(i)$ is the coverage at position i, i.e. the number of fragments that are overlapping position i, j runs over all Gaussian peaks in the model, μ_j and σ_j are the central position and width of the Gaussian j, ρ_j is the fraction of all observations that belong to Gaussian j, and the last term corresponds to the uniform background distribution that accounts for the coverage not associated with the Gaussian peaks. Importantly, for individual binding sites on the genome, the width of the resulting coverage peak is a relatively well-defined function of the fragment length and we use this to constrain the widths σ_j of the Gaussian peaks to fall within a range that is consistent with these peaks corresponding to single binding sites on the genome (see previous section). We also use our knowledge of the typical width of individual binding peaks to constrain the numbers of Gaussians in the mixture model, i.e. we use as many Gaussians as can be fitted within the region given the expected widths of the Gaussians, with a minimum of 2 Gaussians. We again fit the parameters of the model by maximizing the likelihood of the coverage profile using expectation maximisation as follows.

The parameters of equation (2) are vectors, as we are dealing with mixture models of multiple Gaussians. We take the partial derivatives of the log-transformed likelihood with respect to all the parameters, and find the following equations must hold at the maximum:

$$\sigma_j = \sqrt{\frac{\sum_i (i-\mu_j)^2 C_i p_{G_j,i}}{\sum_i C_i p_{G_j,i}}},$$

$$\mu_j = \frac{\sum_i i C_i p_{G_j,i}}{\sum_i C_i p_{G_j,i}},$$

and

$$\rho_j = \frac{\sum_i p_{G_j,i}}{\sum_i C_i},$$

with:

$$p_{G_j,i} = \frac{\frac{\rho_j}{\sqrt{2\pi\sigma_j^2}} \exp(-\frac{(i-\mu_j)^2}{2\sigma_j^2})}{\sum_j \frac{\rho_j}{\sqrt{2\pi\sigma_j^2}} \exp(-\frac{(i-\mu_j)^2}{2\sigma_j^2}) + (1 - \sum_j \rho_j)\frac{1}{l}}.$$

We solve these equations by expectation-maximization, i.e. by iteratively setting new values of σ_j, μ_j and ρ_j using the the above equations, using the old values to compute the expressions on the right-hand sides.

The optimization of the Gaussian mixture may result in some Gaussian components that highly overlap. We merge overlapping Gaussians if the difference between their mean positions is less than the sum of their standard-deviations, which roughly corresponds to the condition that there is no local minimum in the coverage profile between the two peaks. The ρ-weighted averages of the means and standard-deviations of the overlapping Gaussians are used to define the ρ, mean and standard-deviaton of the single merged Gaussian. Finally, for each Gaussian component, the binding peak is defined as the region $[\mu - \sigma, \mu + \sigma]$ (see colored rectangular regions in Fig. 2D of the main text).

Note that, because the number of Gaussians used in the mixture is chosen to be an upper bound on the number of real binding peaks, some of the resulting Gaussian peaks may not exhibit a significant enrichment over background. We thus calculate a z-value for each individual peak and only retain those peaks with a z-value above the threshold z_* computed in the preceding section. The z-value is computed using newly computed counts n which result from summing the contribution of the Gaussian peak in equation (2), together with the uniform background. The renormalized number of reads associated with peak j, in a region of length L, is given by

$$n_j = \frac{\sum_{i=1}^{l} C_i}{f} \left(\rho_j + \frac{W}{l} \left(1 - \sum_k \rho_k \right) \right) \tag{3}$$

where f is the fragment size and W is the window length (500 by default). Note that the first factor corresponds to the total number of fragments assigned to the enriched region. The term proportional to ρ_j corresponds to the fraction of fragments directly assigned to the peak by the mixture model, and the second term corresponds to the total number of background fragments in a region of length W.

1.10 *De Novo* Motif Finding

To identify novel motifs Crunch first uses PhyloGibbs [5], which implements a Bayesian model for assigning posterior probabilities to configurations of putative sites for a number of unknown regulatory motifs (with both the total number of sites and maximum number of different motifs defined by the user), and samples configurations in proportion to their likelihood using Markov chain Monte Carlo. PhyloGibbs was specifically constructed to incorporate information from conservation patterns across orthologous genomic regions given their phylogenetic relationships. For each binding peak Crunch uses UCSCs pairwise genome alignments to extract orthologous sequences from related genomes. When using data from human and mouse CRUNCH uses the hg19, mm9, rheMac2, canFam2, bosTau6, equCab2, and monDom5 asssemblies, and the droSim1, droYak2, droEre2, droAna3, dp4, droWil1, droVir3, droMoj3, and droGri2 assemblies when using data from *Drosophila*. Note that results are insensitive to the precise version of the assembly used. The orthologous regions are then multiply aligned using T-Coffee [6], as described in [7]. To enable the detection of several, potentially non-redundant, motifs we run PhyloGibbs six times with different settings: Either using phylogenetic information and multiple alignments or using only the sequence from the reference species, and searching for motifs of lengths of either 10, 15, or 20 nucleotides. In each case we are searching for two motifs simultaneously (-z 2) defining that both together are expected to have 350 binding sites within the 500 peaks of the training set. Further, we use a first order background model (-N 1). This procedure yields 12 predicted motifs, represented by position specific weight matrices (PWMs). Crunch then refines these PWMs using the MotEvo algorithm [7]. MotEvo uses an expectation maximization procedure to optimize a set of PWMs so as to maximize the likelihood of the input sequences as a mixture of PWM sites and background. Applying this procedure separately to each PWM using the sequences from the training set yields 12 refined motifs. Finally, since these procedures identify motifs of a predefined width, one often observes a core motif flanked by uninformative columns, i.e. columns with nucleotide frequencies matching the background frequencies. Crunch trims all 24 motifs from both ends until a column with information content of at least 0.25 bits appears. Thus, at the end of these procedures, we have at most 24 candidate *de novo* motifs that we will subject to further analysis.

1.11 Library of known motifs

We have collected a large library of known mammalian regulatory motifs from the literature. This library consists of the motif libraries from JASPAR [8], HOCOMOCO [9], HOMER [10], UniPROBE [11], ENCODE [12], HT-SELEX [13], and our own SwissRegulon collection [14], and contains a total of 2325 PWM motifs. For each dataset, we fuse this library of known motifs with the motifs that were found *de novo* for that dataset to form a set of candidate motifs that we denote $\{W_{\text{lib}}\}$.

1.12 Binding site prediction and accounting for non-specific binding

In order to calculate the enrichment score for a set of motifs, we need to determine the number of binding sites $n_{p,\{w\}}$ for the peak sequences as well as for a large pool of background sequences. Although equation (9) of the main text effectively assumes an infinitely large pool of sequences, the score depends only on the average number of binding sites $\langle n_{b,\{w\}} \rangle$ per background sequence, so that the enrichment score can be well estimated provided that the set of background sequences is large enough. We thus created a set of background sequences with 10 times as many sequences as the number of peak sequences. The background sequences are constructed to have the same distribution of lengths and nucleotide composition as the peak sequences. Since, as detailed below, the number of predicted binding sites $n_{p,\{w\}}$ depends on the parameter settings of the binding site prediction algorithm, and the enrichment score additionally depends on the non-specific binding parameter β, we divide the peak and background sequences into two equally sized subsets, P_1 and P_2, where the first 'training' set P_1, containing $\{P_{\text{training}}\}$, is used to optimize β and the parameters of the motif finding, and the second ‘test’ set P_2, containing $\{P_{\text{test}}\}$, is then used to calculate a final enrichment score $E_{\{w\}}$.

We use the MotEvo algorithm [7] to calculate the number of binding sites $n_{p,\{w\}}$ in each sequence p of P_1. MotEvo is a Bayesian algorithm that models the input sequences as a mixture of non-overlapping sites for the motifs from $\{w\}$ and nucleotides deriving from a background model [15] and calculates posterior probabilities of binding sites to occur for each of the motifs in $\{w\}$ at each of the positions in the sequences. MotEvo’s TFBS predictions depend on a set of prior probabilities $\{\pi\}$, with π_w denoting the prior probability that a randomly chosen position in the input sequences corresponds to the start of a binding site for motif $w \in \{w\}$. Crunch runs MotEvo on the training pool P_1 using a mode in which the parameters $\{\pi\}$ are optimized so as to maximize the likelihood of the mixture model on the input data P_1. Once the optimal parameters $\{\pi\}$ are determined, Crunch additionally optimizes the non-specific binding parameter β so as to maximize the enrichment, i.e. equation (9) of the main text, for the training set P_1. The optimal priors $\{\pi^*\}$ and optimal β_* are then fixed, MotEvo is run with these parameters on the test set P_2, and a final enrichment $E_{\{w\}}$ is calculated based on the binding site predictions on the test set.

It is worthwhile to note that the algorithm that MotEvo employs is equivalent to a thermodynamic biophysical model in which the priors $\{\pi\}$ correspond to the concentrations of the TFs associated with the motifs in $\{w\}$ and the posterior probabilities of the sites correspond to the fraction of time the sequence are bound by the respective TFs. In this interpretation the maximization of the priors $\{\pi\}$ corresponds to maximizing the total binding free energy of the input sequences.

Another important point to note is that, as MotEvo only considers non-overlapping configurations of binding sites, redundant motifs compete for binding and consequently will not increase free energy of binding when added to $\{w\}$. More precisely, the sum of the optimized priors of two redundant motifs will be approximately equal to the optimized prior of one of the two redundant motifs by itself. In this way, addition of redundant motifs to the set $\{w\}$ will generally leave the enrichment $E_{\{w\}}$ unchanged.

1.13 Redundant motif removal

The similarity $S(w_1, w_2)$ between motifs w_1 and w_2 is their inner product at their optimal alignment, where the optimal alignment is defined by a shift s that maximizes the inner product $I(s, w_1, w_2)$ and by the orientation of w_2, i.e. also considering the reverse complement $w_{2,rc}$.

$$I(s, w_1, w_2) = \sum_i w_1(i) \cdot w_2(i - s)$$

$$S(w_1, w_2) = \max\left[\max_s[I(s, w_1, w_2)], \max_s[I(s, w_1, w_{2,rc})]\right]$$

Here i runs over all overlapping motif columns. We further normalize $S(w_1, w_2)$ to a number between 0 and 1 and take the difference to 1 to get a dissimilarity measure

$$d(w_1, w_2) = 1 - \frac{2S(w_1, w_2)}{S(w_1, w_1) + S(w_2, w_2)}$$

To now reduce $\{W_{\text{lib}}\}$, which is sorted by enrichment score, we proceed as follows: We move the highest quality motif w_{top} of $\{W_{\text{lib}}\}$, i.e. $w_{\text{top}} = \arg\max_w[E_w \forall w \in \{W_{\text{lib}}\}]$, to a new set $\{W_{\text{reduced}}\}$ and remove all motifs w from $\{W_{\text{lib}}\}$ with $d(w_{\text{top}}, w) < 0.2$. The distance threshold of 0.2 is chosen such that only very close motifs get removed (see Supplementary Figure S15 for examples). This procedure is then repeated until no motif is left in $\{W_{\text{lib}}\}$.

2 Supplemental Text: Biological interpretation of the BRCA1 results

Here we briefly discuss to what extent Crunch's results on BRCA1 are consistent with what's known in the literature. BRCA1 is a tumor suppressor, first discovered in 1990 [16]. BRCA1 is involved in mulitple pathways including DNA damage repair, cell cycle check points, centrosome duplication, and the immune response, and mutated BRCA1 and BRCA2 are reported to be responsible for two thirds of familial breast cancer cases, and also increase risk for ovarian, pancreas, uterus, cervix and prostate cancers [17, 18]. Although BRCA1 is DNA binding and binds to sites of damaged DNA, as part of the BASC complex [19], it does not bind DNA in a sequence-specific manner [20, 18]. The highly enriched denovo_WM_9 thus most likely represents the binding specificity of another TF that associates with BRCA1. Notably, motifs highly similar to denovo_WM_9 have been reported previously. First, a very similar motif occurs primarily in TATA-less promoters, in particular for genes involved in the transition of the G1- to S-phase [21], and BRCA1 has previously been reported to be involved in this transition [22]. Second, denovo_WM_9 is very similar to the UA1 motif found for BRCA1 by ENCODE and to a motif from HOMER's library that describes the binding specificity of the TF ZBTB33, also known as KAISO (Fig. 3D of the main text). KAISO has been associated with breast cancer and especially with BRCA1-related breast cancer [23]. We thus hypothesize that KAISO is a key interaction partner of BRCA1 and that the denovo_WM_9 motif describes the binding specificity of the KAISO TF. Notably, a similar hypothesis was put forward in [12].

Besides denovo_WM_9, Crunch finds five additional motifs that substantially increase the enrichment score and have binding sites within the binding peaks (Fig. 3A of the main text). The most significant of these motifs is a motif associated with the TF CREB3 as inferred from HT-SELEX experiments [13] and Crunch reports that the CREB3 motif is highly similar to the motifs of the ATF and JUN family proteins. BRCA1 is well known to directly bind to CREB-binding protein (CBP) [24] as well as to directly bind to ATF1 [18] and JUN proteins [25]. We thus hypothesize that a complex of BRCA1-CBP-CREB, a complex of BRCA1-ATF1 or a complex of BRCA1-JUN is binding to a subset of our BRCA1 binding peaks.

For the next two motifs, SPI1/PU1 and RFX3, we could not find any support in the literature. Although both together bind only 18 of our peaks, it might still be interesting to further investigate these two motifs in relation to BRCA1. The GFY-staf occurs in almost all peaks and has previously been associated with chromatin regions bound by H3K4me3 which are specifically open in breast cancer cells [26]. Finally, we find the STAT1 motif, a TF that has also been reported to directly bind BRCA1 [18]. In summary, Crunch correctly identifies the key interaction partners of BRCA1 that have previously been described in the literature, and identifies two new motifs that appear to account for BRCA1 binding in a small subset of the peaks.

3 Supplemental Results: Comparison with other peak finders

To compare Crunch's peak predictions with those of MACS2 and SISSR, we created a version of the ChIP-seq pipeline in which Crunch's peak finding was replaced with either MACS2 or SISSR and ran all 128 ENCODE datasets with these two modified pipelines. Both MACS2 and SISSR were run with default settings.

We first compared the extent to which the predicted peaks of the three tools overlap for each dataset. We sorted the peaks by their respective scores and, for each pair of tools and each peak number n calculated the overlap between the top n peaks as the ratio of the intersection of all predicted peak regions divided by the union of all predicted peak regions. As shown in Fig. S5, the overlaps of the Crunch and MACS2 peaks tend to be largest, followed by the overlaps of the Crunch and SISSR peaks, and the MACS2 and SISSR peaks show the least overlap. The overlaps may seem relatively modest, i.e. they typically range from $0.25 - 0.5$ for the top 1000 peaks of Crunch/MACS2 and from $0.15 - 0.3$ for MACS2/SISSR, but it should be noted that this overlap measure is very stringent. For example, if MACS2 consistently predicts peaks that are twice as wide as Crunch's peaks, then the maximum possible overlap is 0.5, even if the two tools predict an identical set of peak centers. Similarly, if all peaks are the same width but are shifted by half a peak-width, the overlap would only be 0.33. Thus, for most datasets the peak sets do show a substantial overlap.

We next compared the predicted peaks for the 31 pairs of datasets in which the same TF was analyzed twice, either in different cells lines or by different labs. For each pair of datasets, and each tool, we calculated the overlap of the top n predicted peak regions for different values of n ranging from 500 top $10'000$. We noted that MACS2 tends to predict wider peaks than Crunch, and SISSR typically predicts narrow peaks. In order to avoid biasing the overlap statistics due to differences in predicted peak widths, we resized the peaks of all three tools to a fixed 200 base pair width, centered on the center of the predicted peak. The overlap between the top n predicted peaks on a pair of 'replicate' datasets was defined as the ratio between the intersection of all predicted regions and the union of all predicted regions. As shown in Fig. S6, Crunch shows consistently the highest overlaps on pairs of datasets for the same TF, independent of the number of peaks in the range $500 - 10'000$. SISSR shows better overlaps than MACS2 for the top 500 and 1000 peaks, but shows worse overlap than MACS2 for the top 5000 and $10'000$ peaks.

As the entire Crunch pipeline was run on all datasets using the MACS2 and SISSR peak finders, motif finding was also performed on the MACS2 and SISSR peaks of each dataset. We first compared, for each dataset, the enrichments of the most enriched motif that the pipeline identified on the peaks of Crunch, MACS2, and SISSR. For example, for the dataset of the transcription factor TCF3 the top motif found for the Crunch peaks was a known motif identified by a HT-SELEX experiment with TCF3 (HTSELEX.TCF3.bHLH.DB.dimeric) which had an enrichment of 4.3 on the Crunch peaks. The most enriched motif on the SISSR peaks was the same HTSELEX.TCF3.bHLH.DB.dimeric motif, but its enrichment on the SISSR peaks was only 2.28. For MACS2 the most enriched motif was a known motif from the HOCOMOCO database, called HCMC.HTF4_f1.wm, which had an enrichment of 2.712. These enrichments are shown as the large blue and orange dots in the left panel of Figure S7. Note that the ratios of the enrichments of these top motifs are $R = 4.3/2.712 \approx 1.59$ when comparing Crunch with MACS2, and $R = 4.3/2.28 \approx 1.89$ when comparing Crunch with SISSR. These ratios R are shown as the blue and orange dots in the right panel of Figure S7. As the reverse cumulative distributions of this figure show, Crunch's top motif exhibits the highest enrichment for roughly 90% of all datasets.

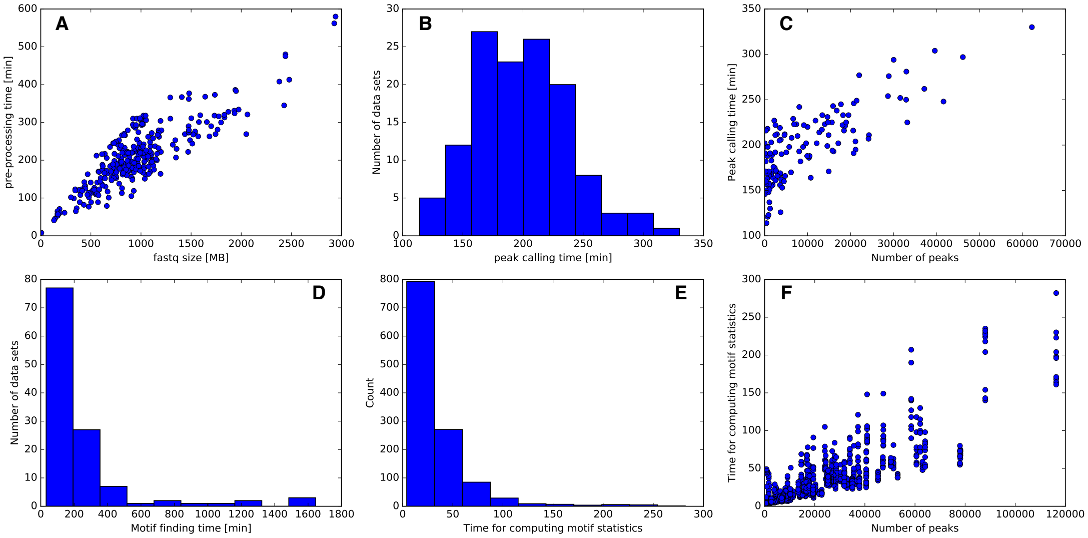
We noted that MACS2 typically predicts significantly wider peaks than Crunch and that SISSR tends to predict very narrow peaks. Since the width of the region over which the ChIP signal shows enrichment depends mainly on the size of the DNA fragments, and is thus not biologically meaningful, we wanted to exclude the possibility that MACS2's and SISSR's poorer performance was due to differences in peak widths, and we thus decided to compare motif enrichment in peak regions of a fixed width. To identify an appropriate width we extracted, for each dataset, the top 1000 peaks as well as the most enriched motif for each of the three tools, and calculated the number of binding sites occurring as a function of position relative to the centers of the peaks. As shown in Fig. 7B of the main paper, all three tools show a clear enrichment of binding sites in a region from -75 to $+75$ relative to the peak center, which corresponds roughly to the amount of DNA covered by a single nucleosome. We thus decided to compare motif enrichments in regions of 150 base pair wide, centered on the centers of all predicted peaks. Second, since different motifs may be optimal for the peak sets of different tools we decided to extract, for each dataset, the most enriched motif for each of the three tools, and to then compare the enrichment of these three motifs on the peak regions of each of the tools. Finally, we also decided to vary the number of top peaks for which the enrichment was calculated from the top 100 peaks to the top $10'000$ peaks. That is, for each dataset, and each n ranging from 100 to $10'000$, we calculated 9 different enrichments:

one for each tool on each of the three top n peak regions of width 150 base pairs. As an example, Supplementary Fig. S8 shows the resulting enrichments for the TCF3 dataset on the top n Crunch peaks (green), MACS2 peaks (blue), and SISSR peaks (orange) as a function of n and using either Crunch's top motif (left panel), MACS2's top motif (middle panel), or SISSR's top motif (right panel). The figure shows that, across almost the entire range of peak numbers, the enrichment of all three top motifs is highest on the Crunch peaks.

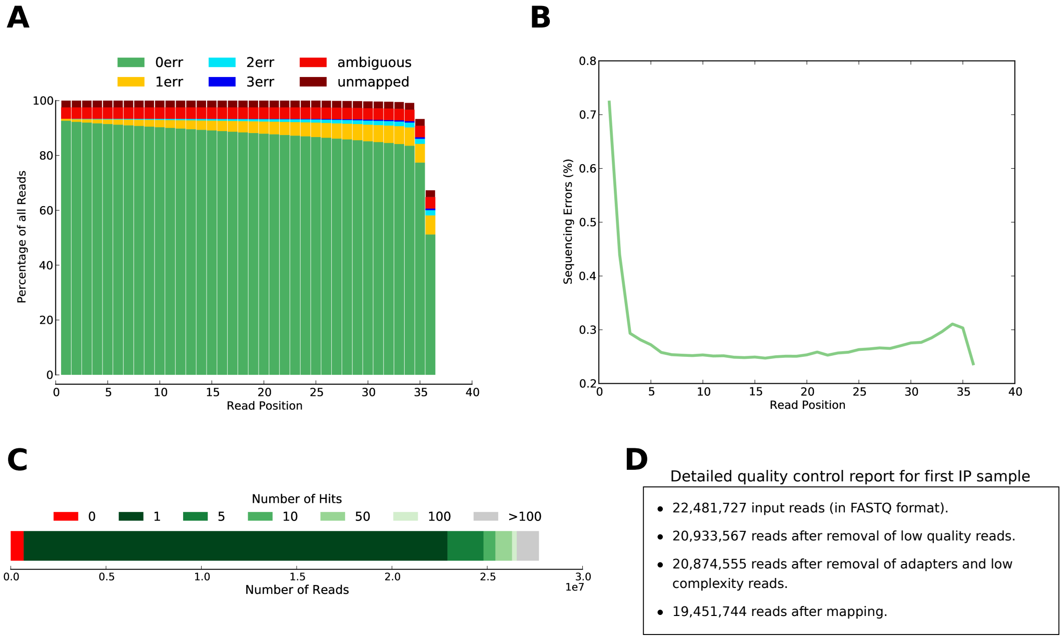
To summarize these enrichment results across all datasets we decided to extract, for each dataset, the enrichments of each of the 3 top motifs on each of the 3 peak sets for the top 200, 500, 1000, 2000, 5000, and top $10'000$ peaks. For each peak number n, and each top motif, we then compared the enrichment of the motif on Crunch's top n peaks with the enrichments of the same motif on the top n peaks of MACS2 and SISSR. In particular, we calculated the ratio R of the enrichment on Crunch's peaks and the enrichment on MACS2's and SISSR's peaks. Finally, for each peak number n we then calculated the distribution of these ratios R across all datasets and the 3 top motifs for each dataset. Fig. S9 shows the reverse cumulative distributions of these enrichment ratios R for Crunch vs MACSs (blue) and Crunch vs. SISSR (orange), with each panel corresponding to a different number of top peaks n. We see that, across all peaks numbers in the range $200 - 10'000$ Crunch achieves the highest motif enrichment for $70 - 80\%$ of all datasets.

Finally, all motif enrichment analysis so far has been based on the enrichment score that we developed, i.e. the average log-ratio of site density and background density across the set of binding peaks of equation (9) of the methods. To verify that our results are not sensitive to this choice of enrichment measure we also performed all enrichment analysis using instead the area under a receiver operator curve, i.e. AUC, as a measure of enrichment. In particular, for each dataset, and each number of top peaks n ranging from $n = 100$ to $n = 1000$ we extracted the top n peak regions for each tool (i.e. regions of 150 bp centered on each peak's center) and created 10-fold more 'background regions' by randomly shuffling the nucleotides within each peak so as to exactly preserve the nucleotide composition of each peak region. Thus, for each n and each of the 3 tools we obtained $11 \times n$ regions of width 150 bp, of which n were true peak regions, and $10 \times n$ were background regions. For each of the 3 top motifs we then predicted binding sites on all $11 \times n$ regions and calculated a ROC curve by using the predicted number of binding sites to classify true peak regions from background regions. The performance of this classification was then measured by the area under the curve (AUC). Note that we thus calculated, for each peak number n in the range 100 to $10'000$, 9 separate AUC values: one for each of the three top motifs on each of the three peak sets. Since the AUC values are all larger than 0.5 and often close to one, we used instead $1 - AUC$ as a measure of the amount of *error* in the classification. Finally, in exact analogy with the analysis of enrichment scores we calculated, for each dataset and each motif, the ratios R of errors on the MACS2 and SISSR peaks versus the errors on the Crunch peaks. That is, an error-rate $R = 2$ corresponds to a case where $1 - AUC$ was half as large on the Crunch peaks as $1 - AUC$ on the peaks of the other tool. Figure S10 shows the reverse cumulative distribution of these errors ratios for Crunch vs MACSs (blue) and Crunch vs. SISSR (orange), with each panel corresponding to a different number of top peaks n. We see that the results based on AUC look extremely similar to the results that we observed with our enrichment score in Fig. S9. That is, also when using AUC as an enrichment measure, Crunch achieves the highest motif enrichment for $70 - 80\%$ of all datasets.

4 Supplementary Figures



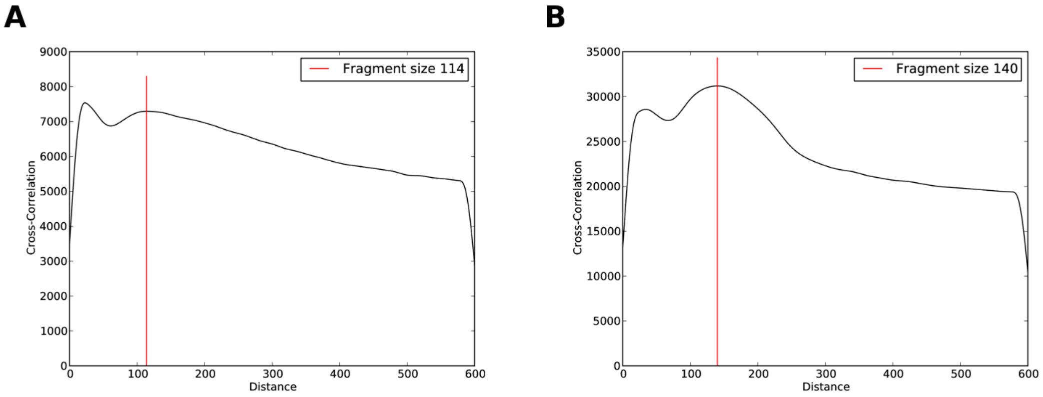
Supplementary Figure S1: **Crunch running times**. **A:** Running time for the pre-processing, i.e. quality control on the reads, read mapping, and fragment size estimation, as a function of the size of the input FASTQ file across the ENCODE datasets. Pre-processing times are roughly 3 hours for a 1 gigabyte file, and 9 hours for a 3 gigabyte file. **B**: Histogram of the time spent in peak-calling, i.e. fitting the statistical model, finding the significantly enriched regions, and decomposing the enriched regions into individual peaks, shows this phase takes typically between 2.5 and 4 hours. **C:** Peak-calling time as a function of the final number of peaks shows that peak-calling times increase with the number of peaks. **D:** Histogram of the time spent on motif finding. For most datasets the motif finding take less than 3 hours, but can take up to a day in rare cases. **E:** Histogram of the time spent on calculating motif statistics for all peaks. Calculating the motif statistics rarely takes more than one hour. **F:** Time spent on calculating motif statistics as a function of the total number of peaks.



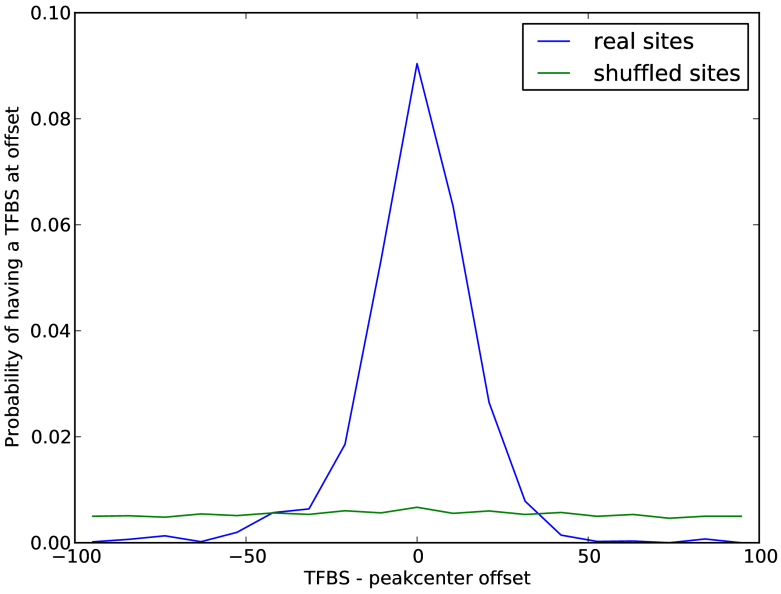
Detailed quality control report for first IP sample

- 22,481,727 input reads (in FASTQ format).
- 20,933,567 reads after removal of low quality reads.
- 20,874,555 reads after removal of adapters and low complexity reads.
- 19,451,744 reads after mapping.

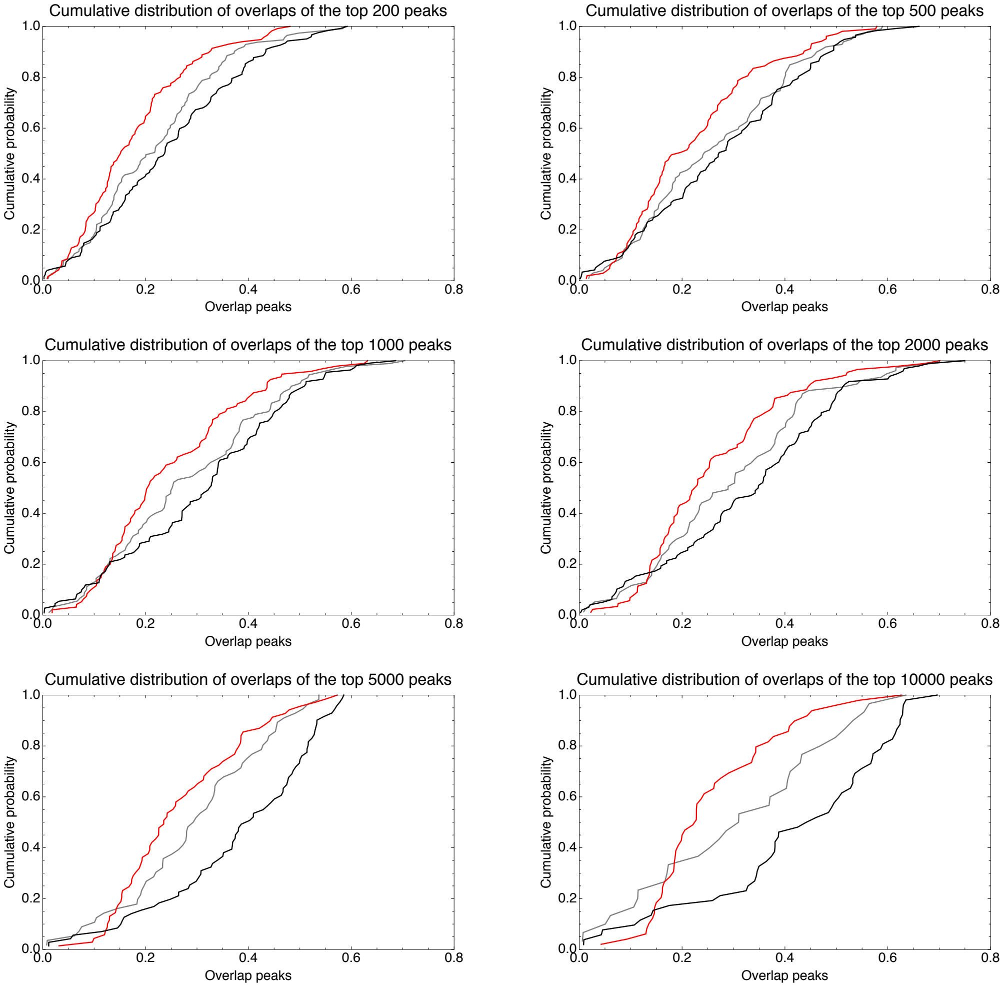
Supplementary Figure S2: **Quality control and read mapping report**. **A**: Cumulative distribution of mapping errors as a function of position in the read. As an example, when considering reads up to read position 20, 85-90% of the reads had no mapping errors (green bar), about 5% had one error (yellow bar), and less than 1% had two or three errors (light and deep blue bars). About 4% were mapped to more than 100 locations (light red bar) and about 2% were not mapped at all (deep red bar). **B**: Percentage of all reads that have a mapping (or sequencing) error as a function of read position. **C**: Cumulative bar plot showing the number of reads with a certain number of mapping locations (hits). All three figures are taken from the second foreground replicate of the BRCA1 dataset from the GM12878 cell-line. **D:** Summary report with the statistics of the quality control and mapping for the first IP sample of the BRCA1 dataset.



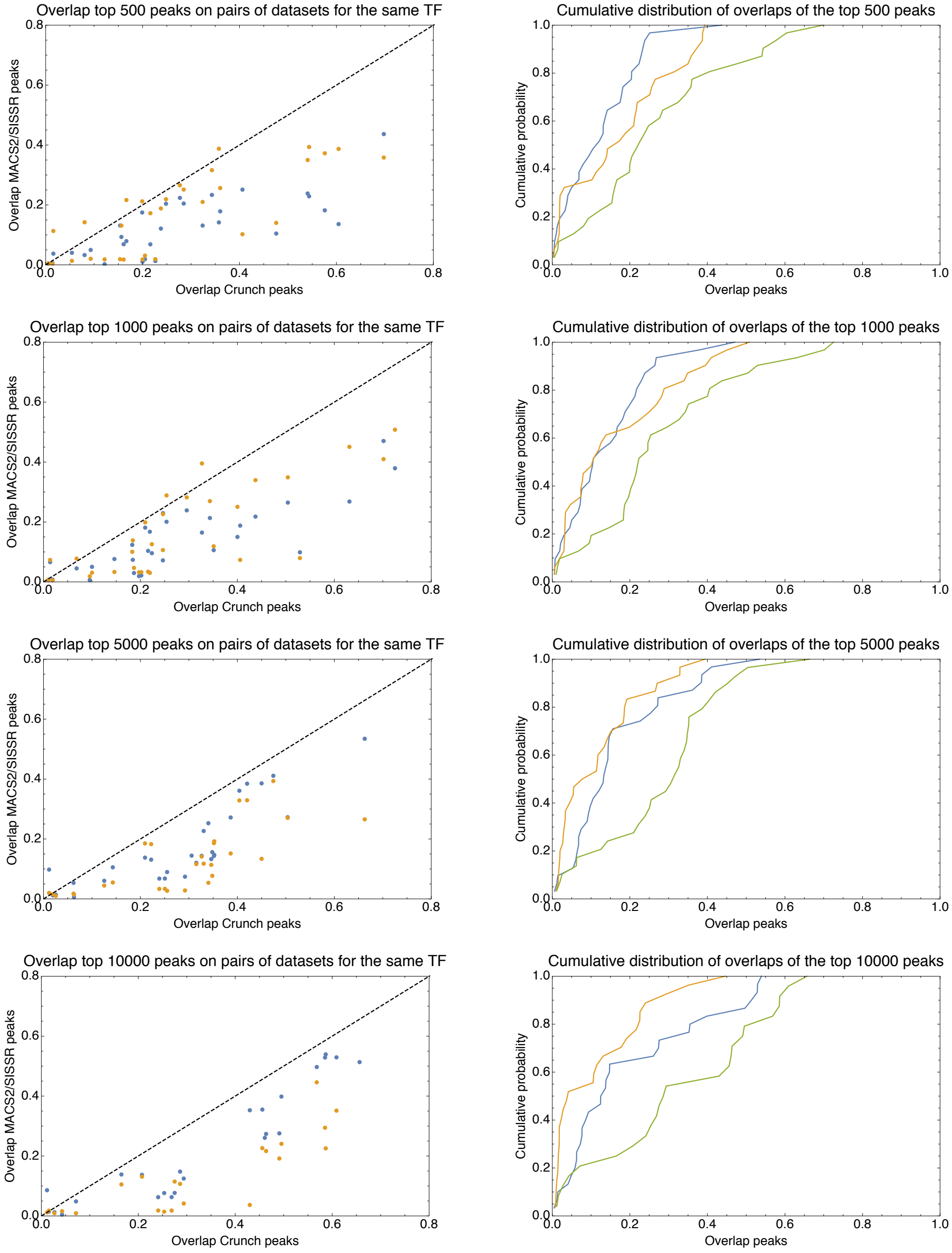
Supplementary Figure S3: **Example cross-correlation profiles**. Cross-correlation $C(d)$ between reads mapped on the plus strand and reads mapped on the minus strand, as a function of their relative strand distance d (see Results section of main text for details) for two example datasets (left: the second ChIP replicate of BAF170, and right: the first ChIP replicate of IRF3, both from the GM12878 cell line). The red vertical line indicates the estimated fragment size. Note that, in both cases, the peak occurring at read length is ignored.



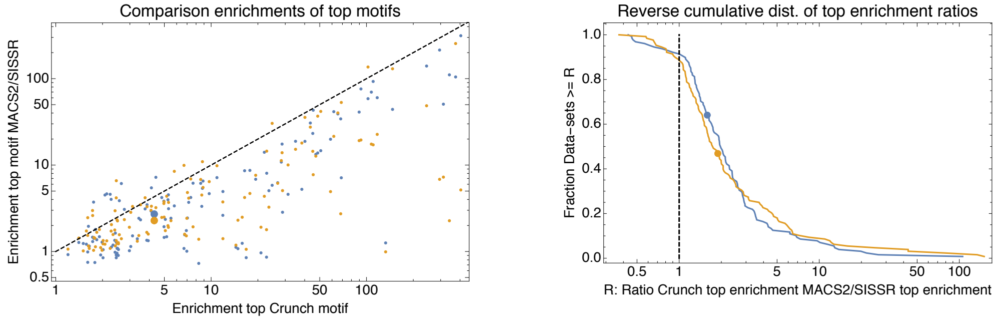
Supplementary Figure S4: **Distances between peak centers and motif occurrence**. A histogram of the offset between peak centers of the BRCA1 peaks and the nearest predicted binding site of the top scoring motif (denovo_WM_9). Note that a binding site typically occurs within 20 bps of the peak's center. The green line shows the corresponding histogram for a control in which all denovo_WM_9 binding sites have been shuffled within the peak sequences in which they occur, showing that the enrichment at peak centers is not just due to a high density of denovo_WM_9 sites.



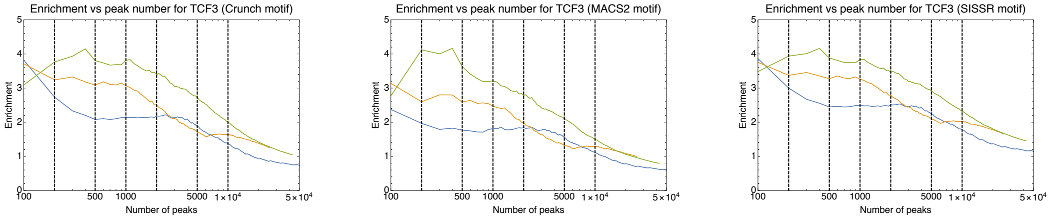
Supplementary Figure S5: **Cumulative distributions of overlaps between the peaks of Crunch, MACS2 and SISSR**. For each dataset, each pair of tools, and a given peak number n, we calculated the overlap of the predicted peaks, i.e. the ratio between the intersection and the union of the top n peak regions. The figure shows the cumulative distribution of overlaps with each panel corresponding to a peak number n, ranging from 200 to $10'000$ and the overlaps of Crunch and MACS2 shown in black, the overlaps of Crunch and SISSR shown in gray, and the overlaps of MACS2 and SISSR shown in red. We see that MACS2 and SISSR consistently show the lowest overlaps in predicted peaks.



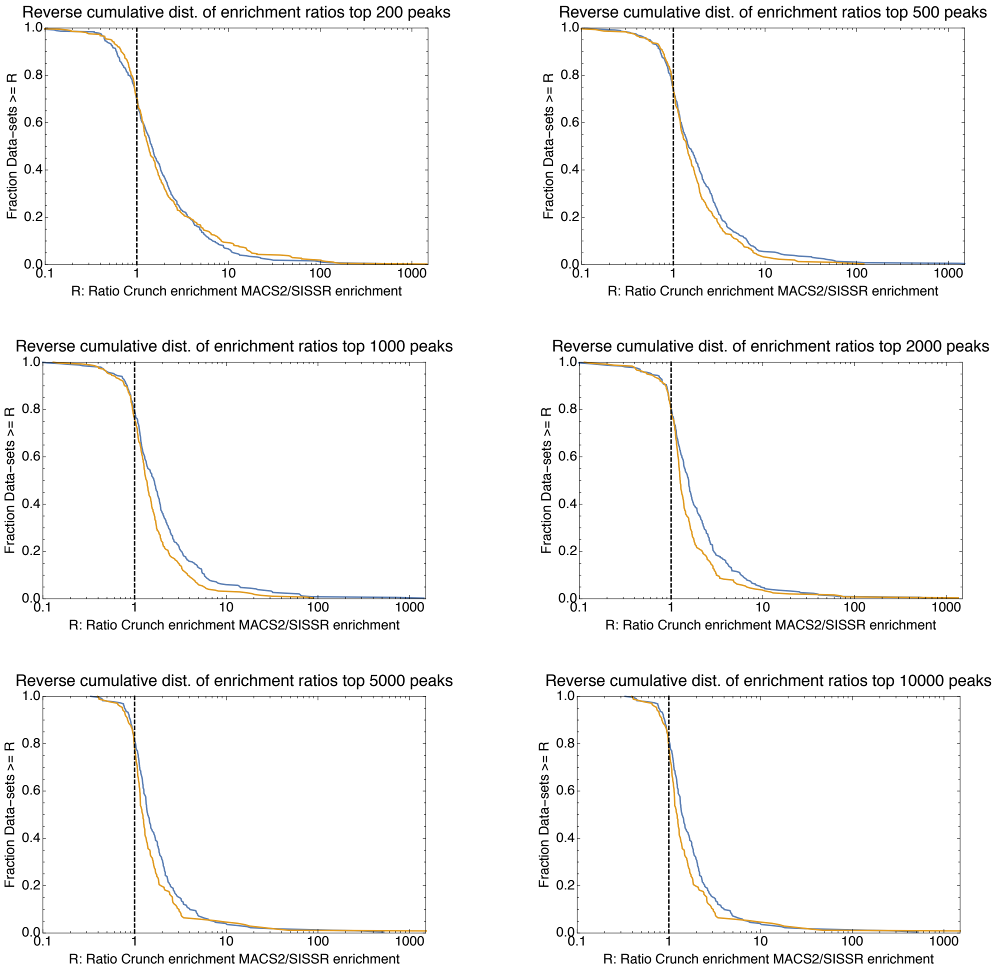
Supplementary Figure S6: **Overlaps of predicted peaks on pairs of datasets for the same TF**. For each pair of datasets in which the same TF was analyzed twice, and each of the three tools, we calculated the overlap of the top n peak regions across the pair of datasets. The left panels show the overlaps of the Crunch peaks (horizontal axis) against the overlaps of the MACS2 (blue) and SISSR (orange) peaks for the same pair of replicate datasets, with the dashed line showing the line $y = x$. The fact that almost all points are below the diagonal shows that Crunch's peaks consistently show higher overlap on these 'replicate' datasets. The right panels show the cumulative distribution of overlaps across all pairs of 'replicate' datasets for Crunch (green), MACS2 (blue), and SISSR (orange). Each row of panels corresponds to a different number n of top peaks ranging from 500 (top row) to $10'000$ bottom row.



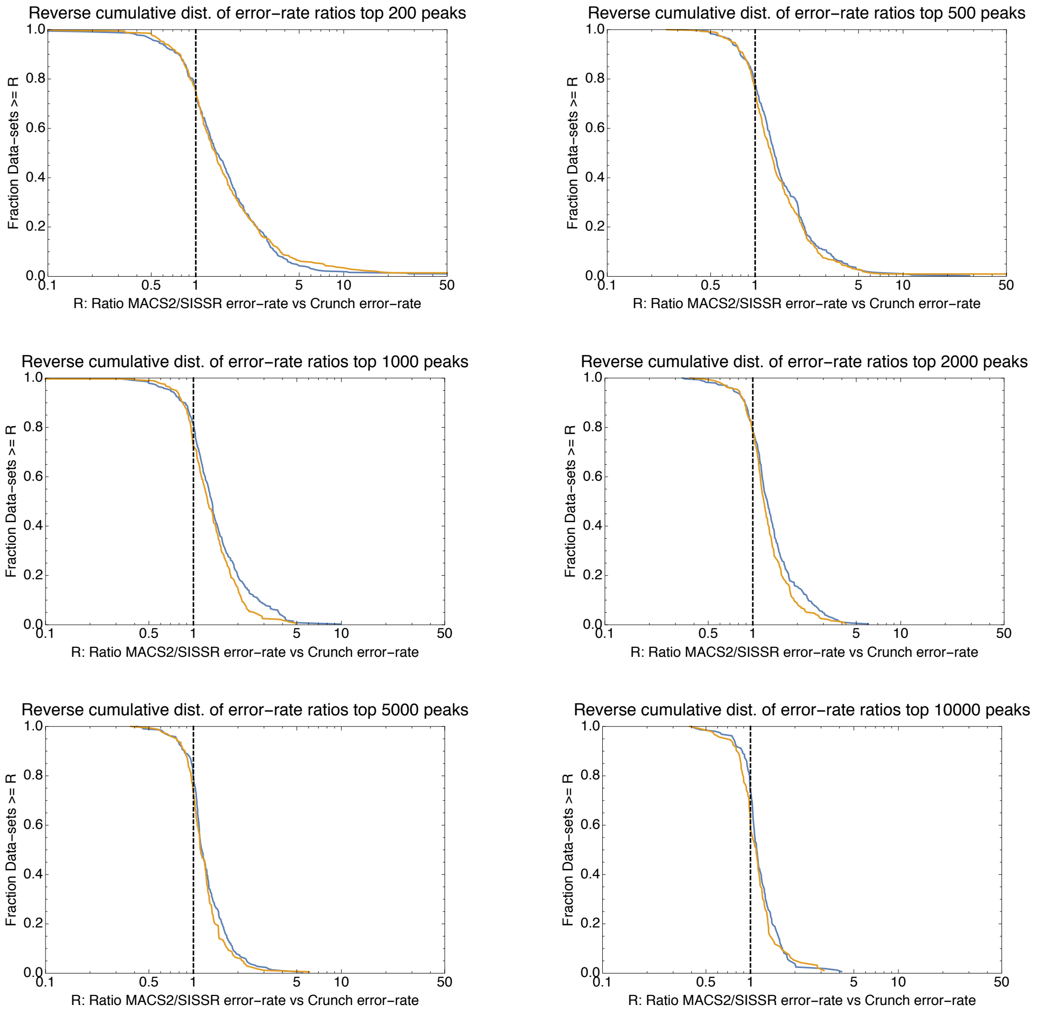
Supplementary Figure S7: **Comparison of the enrichments of the top enriched motifs for Crunch, MACS2, and SISSR**. For each dataset the Crunch pipeline identified the most enriched motif on the top 1000 peaks of Crunch, MACS2, and SISSR. The left panel shows the enrichment of Crunch's top motif (horizontal axis) against the enrichment on the same dataset of the top motif of MACS2 (blue) and SISSR (orange). The right panel shows the distribution of the ratios R of the enrichment of Crunch's top motif and the enrichment of the top motif on the same data for MACS2 (blue) and SISSR (orange). The dashed line indicates a ratio of $R = 1$, i.e. to the right of the dashed line Crunch's top motif had higher enrichment than the top motif of MACS2/SISSR. Note that Crunch's motif had highest enrichment for approximately 90% of the dataset. The thick dots in both panels indicate the enrichments on the example TCF3 dataset.



Supplementary Figure S8: **Enrichment of the top motifs on the TCF3 dataset**. The figure shows the enrichment of the top motif for Crunch (left panel), MACS2 (middle panel), and SISSR (right panel) on the top 150 bp wide peak regions of Crunch (green), MACS2 (blue), and SISSR (orange), as a function of the number n of top peaks (horizontal axis). The vertical dashed lines at $200, 500, 1000, 2000, 5000$, and $10'000$ show the peak numbers for which we calculated the distributions of enrichment ratios shown in the next figure.



Supplementary Figure S9: **Comparison of motif enrichment ratios**. For each dataset and each of the 3 top motifs, we calculated the ratio R of enrichments of the motif on the top n Crunch peaks versus the enrichment on the top n MACS2, as well as the ratio R of enrichments on the Crunch peaks versus the SISSR peaks. Again peaks of fixed 150 bp width were used. Each panel shows the reverse cumulative distribution of enrichment ratios R for Crunch versus MACS2 (blue) and Crunch versus SISSR (orange) for a different value of the top peak number n, ranging from $n = 200$ (top left) to $n = 10'000$ (bottom right). The vertical dashed lines correspond to $R = 1$, i.e. to the right of the dashed line Crunch's peaks show higher enrichment than those of MACS2 and SISSR. Note that, across all peak numbers, Crunch's peaks exhibit highest enrichment for $70 - 80\%$ of all datasets.


Reverse cumulative dist. of error−rate ratios top 200 peaks

Reverse cumulative dist. of error−rate ratios top 500 peaks

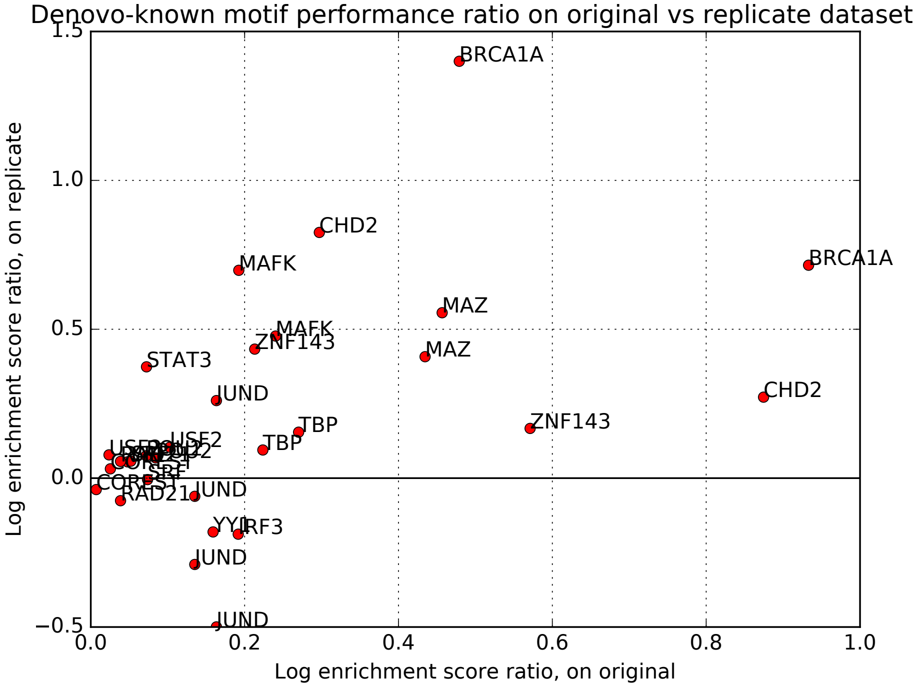
Reverse cumulative dist. of error−rate ratios top 1000 peaks

Reverse cumulative dist. of error−rate ratios top 2000 peaks

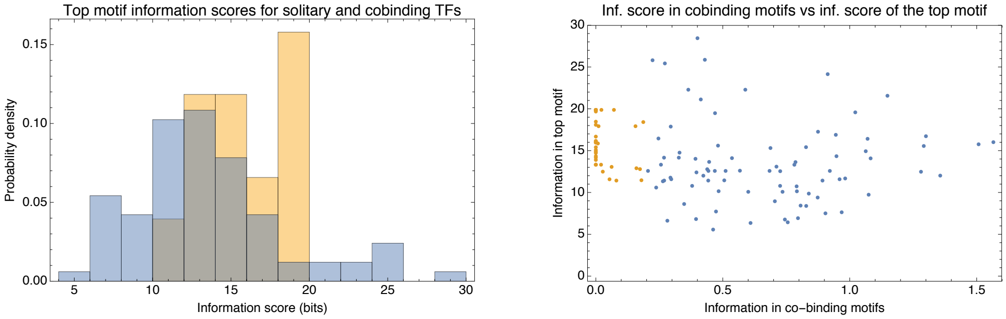
Reverse cumulative dist. of error−rate ratios top 5000 peaks

Reverse cumulative dist. of error−rate ratios top 10000 peaks

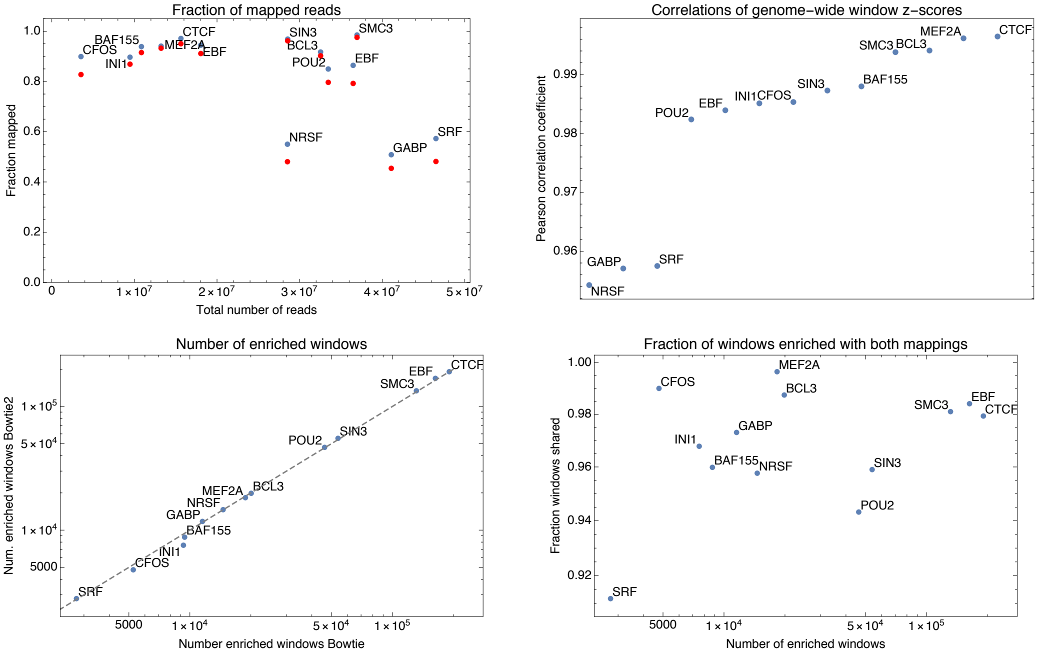
Supplementary Figure S10: **Comparison of motif enrichment ratios based on AUC scores**. For each dataset and each of the 3 top motifs, we calculated the ratio R of the error-rate $(1-AUC)$ of classification of the motif on the top n peaks of MACS2 versus Crunch and SISSR versus Crunch. Again, peaks of fixed 150 bp width were used. Each panel shows the reverse cumulative distribution of error-rate ratios R for MACS2 versus Crunch (blue) and SISSR versus Crunch (orange) for a different value of the top peak number n, ranging from $n = 200$ (top left) to $n = 10'000$ (bottom right). The vertical dashed lines correspond to $R = 1$, i.e. to the right of the dashed line Crunch's peaks show less error in classification than those of MACS2 and SISSR. Note that, across all peak numbers, Crunch's peaks exhibit a lower error-rate for $70 - 80\%$ of all datasets.



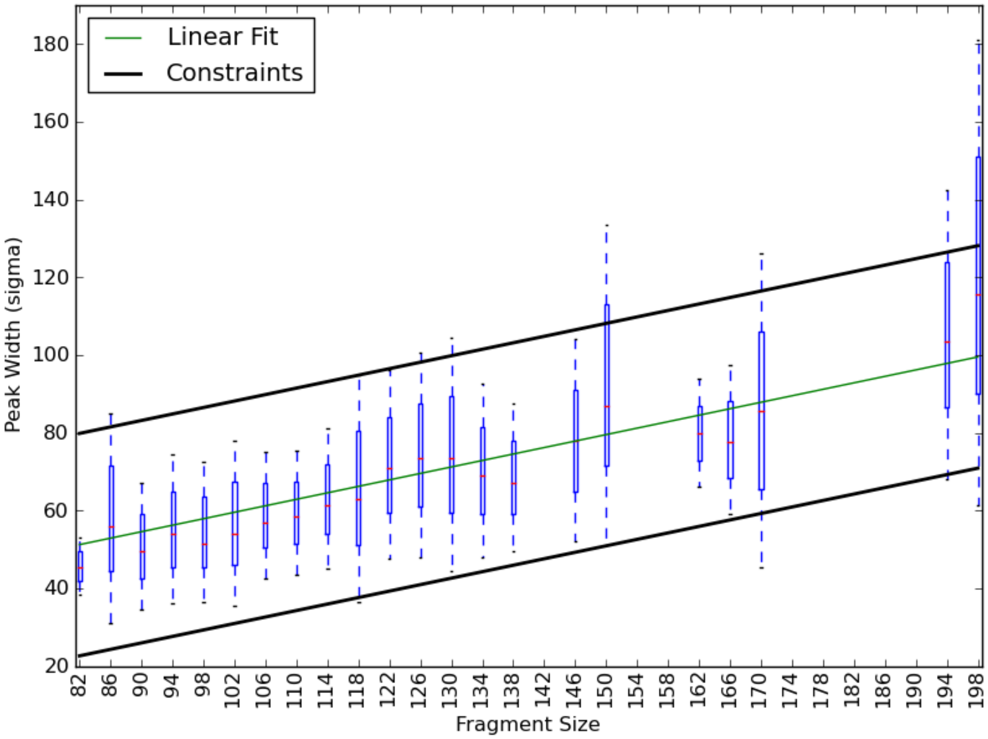
Supplementary Figure S11: **Enrichment ratios between best *de novo* and known motifs on pairs of datasets for the same TF**. For each dataset for which the best motif M_d was found *de novo*, and for which a second dataset was available for the same TF, we extracted the top known motif M_k and calculated the log-ratio of enrichments for the motifs M_d and M_k both on the original dataset (horizontal axis), and on the 'replicate' dataset (vertical axis). The name of each datasets' TF is shown next to the point. We see that for the large majority of cases (i.e. 22 out of 30) the *de novo* motif also outcompetes the best known motif on the replicate dataset.



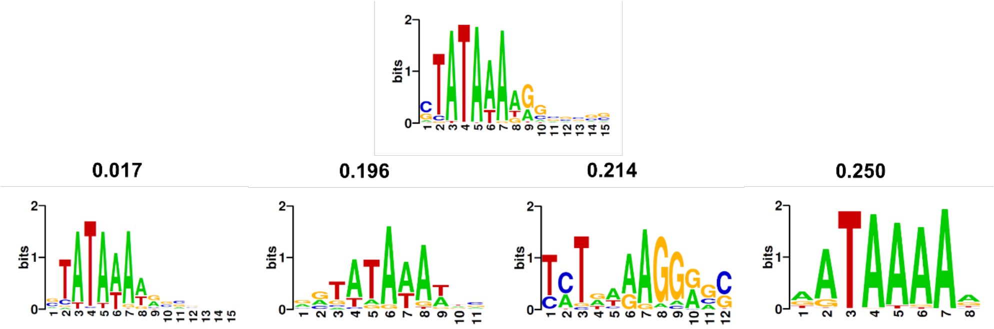
Supplementary Figure S12: **Information scores of the top motifs for solitary and co-binding TFs**. For each TF we calculated the information score of the PWM of its top motif in bits. **Left panel**: Distribution (probability density) of the information scores of solitary (orange) and co-binding (blue) TFs. **Right panel**: Information score of the top motif as a function of the additional information contained in the co-binding motifs. The solitary TFs are shown in orange and the co-binding TFs in blue. Note that there is no noticeable correlation between the information score of the top motif and the additional information that is contained in the co-binding motifs.



Supplementary Figure S13: **Comparison of Bowtie and Bowtie2 mappings**. We randomly selected 10% of the ENCODE datasets and replaced Bowtie with the newer verson Bowtie2 in the mapping step. **Top left:** The total fraction of mapped reads as a function of the total number of reads for Bowtie (blue) and Bowtie2 (red). The TF that was immunoprecipitated in each dataset is indicated next to the corresponding datapoint. Bowtie2 maps up to 10% less reads than the original Bowtie. **Top right:** Pearson correlation of the ChIP enrichment z-scores in 500 bp windows genome-wide. The correlations are over 0.98 except for 3 datasets with low mapping rates, which have correlations just below 0.96. **Bottom left:** Total number of significantly enriched windows when using Bowtie (horizontal axis) or Bowtie2 (vertical axis) mappings. The dotted line corresponds to $y = x$. Note that, even though Bowtie2 maps less reads, the number of significantly enriched windows is virtually identical. **Bottom right:** Overlap of the set of significantly enriched windows obtained with the Bowtie and Bowtie2 mappings as a function of the total number of enriched windows. Overlaps are typically around 98%.



Supplementary Figure S14: **Peak widths as a function of fragment size.** The boxplots (blue) and median (red) of the distribution of peak widths from 123 ENCODE ChIP-seq experiments as a function of the estimated fragment size. The green line shows a linear regression fit to the data. The bold black lines indicate the boundaries within which peak widths are constrained during the mixture modeling of enriched regions.



Supplementary Figure S15: **Motif distance measure**. On top, the sequence motif logo of the TATA-box binding protein (TBP) from SwissRegulon is shown and below it four similar motifs are shown corresponding to from left to right, TBP from JASPAR, TBP from HOCOMOCO, GTF2A1,2 from SwissRegulon and CPEB1 from HTSELEX. For each motif at the bottom, the distance to the top motif is shown. Note that a distance of 0.2 was chosen as the threshold for calling motifs highly similar.

References

[1] Illumina incorporated. Illumina adapter sequences (2018). Https://support.illumina.com/downloads/illumina-customer-sequence-letter.html.

[2] Stadler, M. B. *et al.* DNA-binding factors shape the mouse methylome at distal regulatory regions. *Nature* **480**, 490–5 (2011). URL http://www.ncbi.nlm.nih.gov/pubmed/22170606.

[3] Langmead, B., Trapnell, C., Pop, M. & Salzberg, S. L. Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. *Genome biology* **10**, R25 (2009). URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2690996{\&}tool=pmcentrez{\&}rendertype=abstract.

[4] Landt, S. G. *et al.* ChIP-seq guidelines and practices of the ENCODE and modENCODE consortia. *Genome Res.* **22**, 1813–1831 (2012).

[5] Siddharthan, R., Siggia, E. D. & van Nimwegen, E. PhyloGibbs: a Gibbs sampling motif finder that incorporates phylogeny. *PLoS computational biology* **1**, e67 (2005). URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1309704{\&}tool=pmcentrez{\&}rendertype=abstract.

[6] Notredame, C., Higgins, D. G. & Heringa, J. T-Coffee: A novel method for fast and accurate multiple sequence alignment. *Journal of molecular biology* **302**, 205–17 (2000). URL http://www.ncbi.nlm.nih.gov/pubmed/10964570.

[7] Arnold, P., Erb, I., Pachkov, M., Molina, N. & van Nimwegen, E. MotEvo: integrated Bayesian probabilistic methods for inferring regulatory sites and motifs on multiple alignments of DNA sequences. *Bioinformatics (Oxford, England)* **28**, 487–94 (2012). URL http://www.ncbi.nlm.nih.gov/pubmed/22334039.

[8] Bryne, J. C. *et al.* JASPAR, the open access database of transcription factor-binding profiles: new content and tools in the 2008 update. *Nucleic acids research* **36**, D102–6 (2008). URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2238834{\&}tool=pmcentrez{\&}rendertype=abstract.

[9] Kulakovskiy, I. V. *et al.* HOCOMOCO: a comprehensive collection of human transcription factor binding sites models. *Nucleic acids research* **41**, D195–202 (2013). URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3531053{\&}tool=pmcentrez{\&}rendertype=abstract.

[10] Heinz, S., Benner, C., Spann, N. & Bertolino, E. Simple Combinations of Lineage-Determining Transcription Factors Prime cis-Regulatory Elements Required for Macrophage and B Cell Identities. *Molecular cell* **38**, 576–589 (2010). URL http://www.sciencedirect.com/science/article/pii/S1097276510003667.

[11] Newburger, D. E. & Bulyk, M. L. UniPROBE: an online database of protein binding microarray data on protein-DNA interactions. *Nucleic acids research* **37**, D77–82 (2009). URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2686578{\&}tool=pmcentrez{\&}rendertype=abstract.

[12] Wang, J., Zhuang, J., Iyer, S. & Lin, X. Sequence features and chromatin structure around the genomic regions bound by 119 human transcription factors. *Genome …* 1798–1812 (2012). URL http://genome.cshlp.org/content/22/9/1798.short.

[13] Jolma, A. *et al.* DNA-Binding Specificities of Human Transcription Factors. *Cell* **152**, 327–339 (2013). URL http://linkinghub.elsevier.com/retrieve/pii/S0092867412014961.

[14] Pachkov, M., Balwierz, P. J., Arnold, P., Ozonov, E. & van Nimwegen, E. SwissRegulon, a database of genome-wide annotations of regulatory sites: recent updates. *Nucleic acids research* **41**, D214–20 (2013). URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3531101{\&}tool=pmcentrez{\&}rendertype=abstract.

[15] van Nimwegen, E. Finding regulatory elements and regulatory motifs: a general probabilistic framework. *BMC bioinformatics* **8 Suppl 6**, S4 (2007). URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1995539{\&}tool=pmcentrez{\&}rendertype=abstract.

[16] Hall, J., Lee, M., Newman, B. & Morrow, J. Linkage of early-onset familial breast cancer to chromosome 17q21. *Science* **250**, 17–22 (1990). URL http://www.sciencemag.org/content/250/4988/1684.short.

[17] Deng, C. X. & Brodie, S. G. Roles of BRCA1 and its interacting proteins. *Bioessays* **22**, 728–737 (2000).

[18] Rosen, E., Fan, S. & Ma, Y. BRCA1 regulation of transcription. *Cancer letters* **236**, 175–185 (2006). URL http://www.sciencedirect.com/science/article/pii/S0304383505004519.

[19] Wang, Y., Cortez, D. & Yazdi, P. BASC, a super complex of BRCA1-associated proteins involved in the recognition and repair of aberrant DNA structures. *Genes & ...* 927–939 (2000). URL http://genesdev.cshlp.org/content/14/8/927.short.

[20] Starita, L. The multiple nuclear functions of BRCA1: transcription, ubiquitination and DNA repair. *Current Opinion in Cell Biology* **15**, 345–350 (2003). URL http://linkinghub.elsevier.com/retrieve/pii/S0955067403000425.

[21] Wyrwicz, L. S., Gaj, P., Hoffmann, M., Rychlewski, L. & Ostrowski, J. A common cis-element in promoters of protein synthesis and cell cycle genes. *Acta biochimica Polonica* **54**, 89–98 (2007). URL http://www.ncbi.nlm.nih.gov/pubmed/17351670.

[22] Fabbro, M. *et al.* BRCA1-BARD1 complexes are required for p53Ser-15 phosphorylation and a G1/S arrest following ionizing radiation-induced DNA damage. *The Journal of biological chemistry* **279**, 31251–8 (2004). URL http://www.ncbi.nlm.nih.gov/pubmed/15159397.

[23] Vermeulen, J. F. *et al.* Nuclear Kaiso expression is associated with high grade and triple-negative invasive breast cancer. *PloS one* **7**, e37864 (2012). URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3360634{\&}tool=pmcentrez{\&}rendertype=abstract.

[24] Pao, G. M., Janknecht, R., Ruffner, H., Hunter, T. & Verma, I. M. CBP/p300 interact with and function as transcriptional coactivators of BRCA1. *Proc. Natl. Acad. Sci. U.S.A.* **97**, 1020–1025 (2000).

[25] Hu, Y. & Li, R. JunB potentiates function of BRCA1 activation domain 1 (AD1) through a coiled-coil-mediated interaction. *Genes & development* **1**, 1509–1517 (2002). URL http://genesdev.cshlp.org/content/16/12/1509.short.

[26] Hong, C. P., Choe, M. K. & Roh, T.-Y. Characterization of Chromatin Structure-associated Histone Modifications in Breast Cancer Cells. *Genomics & informatics* **10**, 145–52 (2012). URL `http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3492650{\&}tool=pmcentrez{\&}rendertype=abstract`.