ZINBRA: ChIP-seq enrichment detection and comparison tool

Aleksei Dievskii

September 23, 2015

- Detect enriched regions in ChIP-seq data (one or more replicates)
- Detect differentially enriched regions in ChIP-seq data (one or more replicates for each of two tracks)
- Use both features to explore the epigenetic dynamics of differentiation and ageing

- Input: a library of ChIP-seq tags
  (unique reads alignment, redundant reads eliminated)
- Each chromosome is tiled with bins of fixed length
  (default: 200 bp)
- The tags are aggregated for each bin (binned coverage),
  producing an array of integers for each replicate
  and chromosome

The integer series are used to fit a 3-state multidimensional negative binomial HMM:

- NULL state: singular distribution $d_i = 0$ for each dimension i
- LOW state: $d_i \sim NB(m_i^-, f_i^-)$
- HIGH state: $d_i \sim NB(m_i^+, f_i^+)$

The parameters are fitted by the usual Baum-Welch EM algorithm. HIGH state corresponds to enriched regions.

The integer series are used to fit a 5-state multidimensional negative binomial HMM:

- NULL state: singular distribution $d_{1,i} = 0$, $d_{2,j} = 0$ for each dimension i, j
- LOW state: $d_{1,i} \sim NB(m^-_{1,i}, f^-_{1,i})$, $d_{2,j} \sim NB(m^-_{2,j}, f^-_{2,j})$
- INC state: $d_{1,i} \sim NB(m^-_{1,i}, f^-_{1,i})$, $d_{2,j} \sim NB(m^+_{2,j}, f^+_{2,j})$
- DEC state: $d_{1,i} \sim NB(m^+_{1,i}, f^+_{1,i})$, $d_{2,j} \sim NB(m^-_{2,j}, f^-_{2,j})$
- HIGH state: $d_{1,i} \sim NB(m^+_{1,i}, f^+_{1,i})$, $d_{2,j} \sim NB(m^+_{2,j}, f^+_{2,j})$

INC and DEC states correspond to differentially enriched regions.

We determine the membership probabilities (e.g. $P(s_t = \mathrm{HIGH})$) for each bin and apply the Q-value FDR control procedure to eliminate false positives. The bin size and the FDR α value are the only parameters of the algorithm.

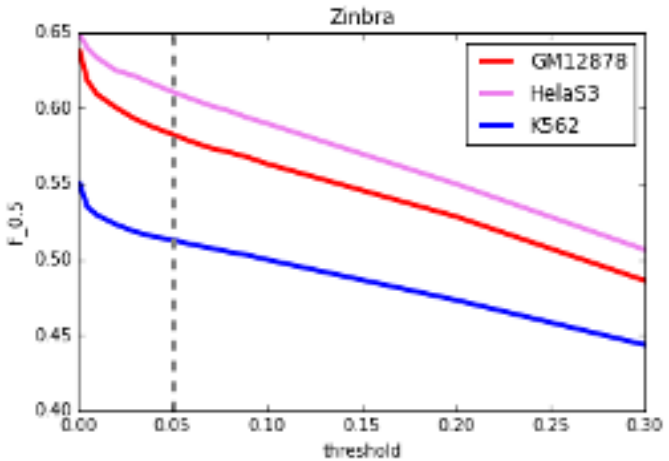
- Written in Java and Kotlin as a part of a larger framework
- Currently available as a standalone JAR
- Chromosome-level concurrency (multi-threaded)
- SIMD optimization (SSE2, AVX)
- Less than five minutes per chromosome when run on a desktop

We began with a Poisson model, but switched to NB.

1. The real data don't quite look like Poisson: the (weighted) mean is much less than the (weighted) variance;
2. NB produces a much better fit, as evidenced by AIC and BIC;
3. NB offers a much more gentle state separation;
4. Poisson is a marginal case of NB ($f \to \infty$);
5. NB is neatly interpretable as a Poisson-Gamma mixture.

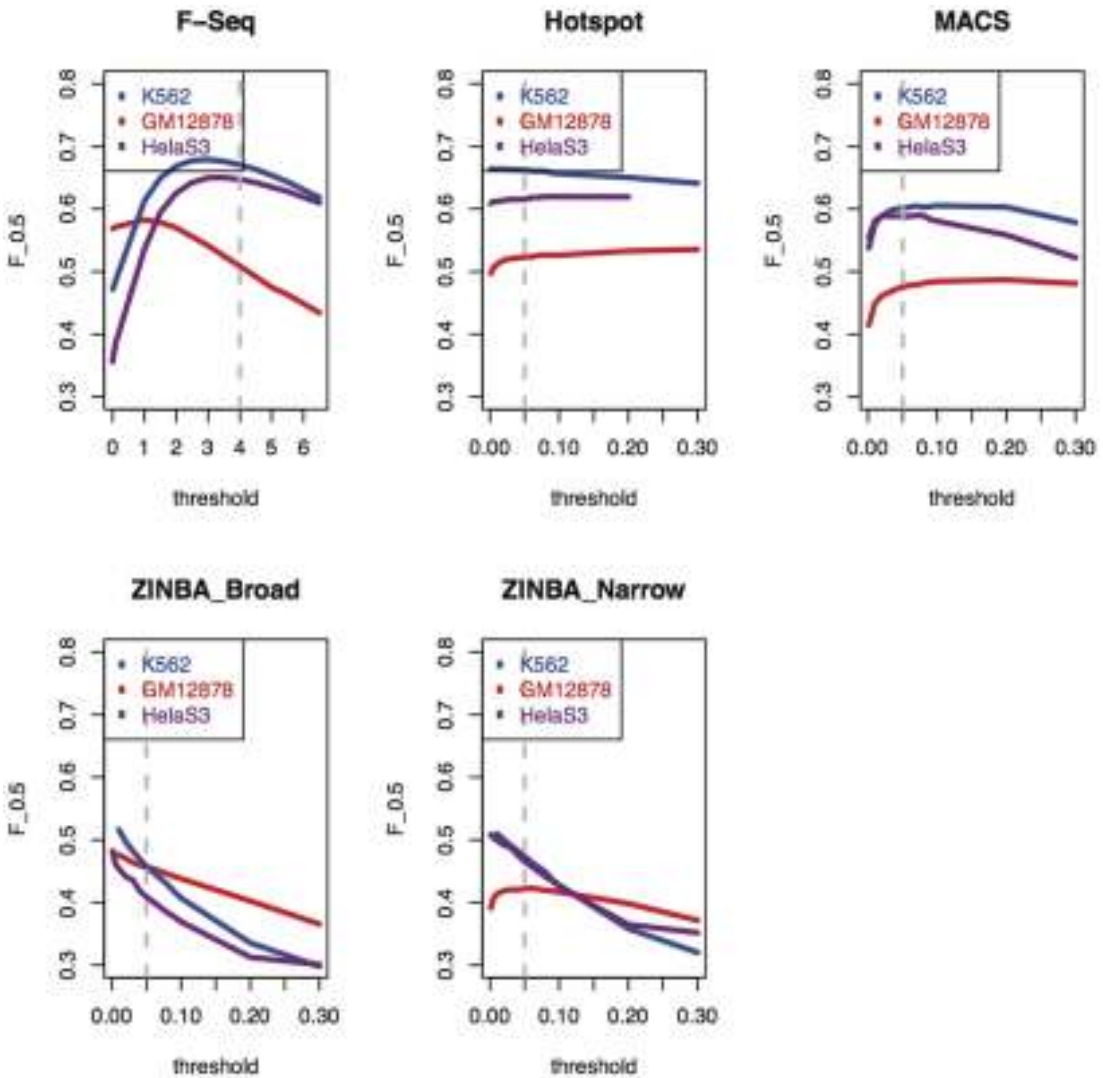
Validation is somewhat difficult, since we generally don't know "the correct answer."
The model performs expectedly well on simulated data (sampling and re-learning); this is actually covered by JUnit tests.

We reproduced an experiment comparing several peak callers based on their recovery of open chromatin data (DNAase-seq instead of ChIP-seq, but close enough). Our model performs remarkably well at low FDR α levels even at one replicate. None of the other callers seem to be capable of accepting multiple ones at all.



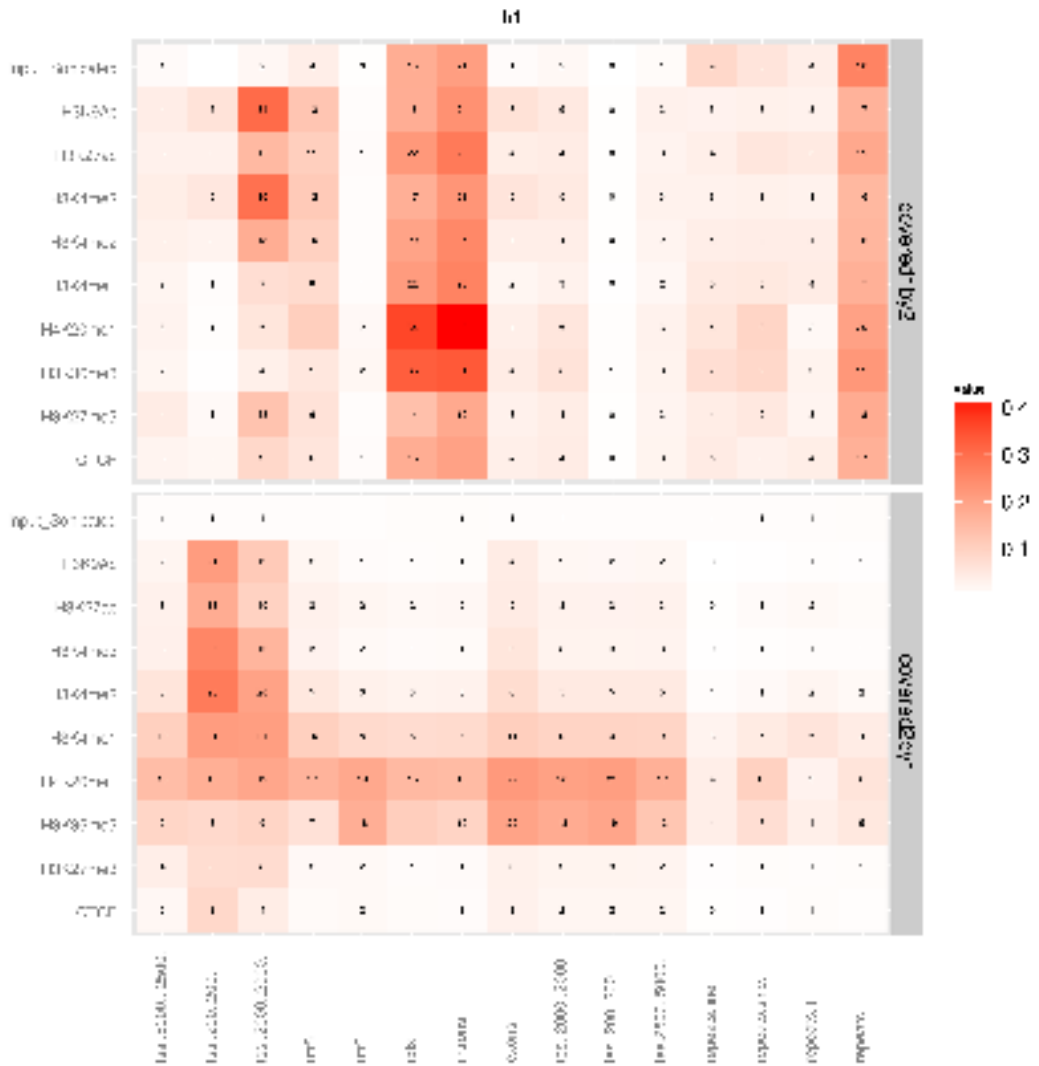
$$\left(F_\beta = (1 + \beta^2) \cdot \frac{\text{precision} \cdot \text{recall}}{\beta^2 \cdot \text{precision} + \text{recall}} \right)$$

Validation: real data[1]



[1] A Comparison of Peak Callers Used for DNase-Seq Data, Hashem Koohy, Thomas A. Down, Mikhail Spivakov, Tim Hubbard, 2014

Validation: real data[2]



[2]GSE26320: Mapping and analysis of chromatin state dynamics in nine human cell types

- MACS (Model-assisted analysis of ChIP-seq) attempts to compensate for the tag shift (the peak is usually near the center of the fragment, while the tag represents its head). Depends on several magic constants (e.g. minimum fold enrichment), no underlying generating model.
- SICER (a clustering approach for identification of enriched domains from histone modification ChIP-Seq data) searches for islands or clusters and doesn't offer a clear way of telling whether the given region is enriched. Also depends on magic constants (e.g. gap size), also no underlying generating model.

Why invent a bicycle?

- We needed a peak caller integrated into our framework
- We couldn't find any sensible peak callers that were able to work with replicates
- We needed a comparison method as well

- ZINBRA
- JetBrains Research: BioLabs
- Genome Browser
- MACS
- SICER
- Peak caller F-score comparison (original article)
- Peak caller F-score comparison (Python notebook)

Thank you for your attention!