Using peakC with capC-MAP output

Chris Brackley

19 November 2018

peakC is an R tool for finding significant interactions in Capture-C and 4C data, available at https://github.com/deWitLab/peakC, and detailed in the paper

> Geeven, G., Teunissen, H., de Laat, W., and de Wit, E. "peakC: a flexible, non-parametric peak calling package for 4C and Capture-C data" *Nucleic acids research* **46** (2018) e91.

which is available at https://doi.org/10.1093/nar/gky443.

Output from capC-MAP can be loaded into peakC provided it is first converted to 'wig' file format. This can be done with, for example the following Unix command:

```
awk '{
if (NR==1) {
   if ($1=="track") {$2="type=wiggle_0"; print}
   else {print "track type=wiggle_0"}
}
span=$3-$2;
chr=$1;
if ($1!="track") {
   if (chr!=lastchr || span!=lastspan) {
      print "variableStep chrom=" chr " span=" span
   }
   print $2,$4
}
lastspan=span;
lastchr=chr;
}' data1/captured_bin_200_2000_RPM_Pax6.bdg > captured_bin_200_2000_RPM_Pax6.wig
```

(and see also the "file_conversion" tutorial).

Then the following R commands are used to call peaks for a target from a single experiment:

```
library(peakC)

viewpoint <- 105521790

data <- readqWig("captured_bin_200_2000_RPM_Pax6.wig", vp.pos=viewpoint, window=700e3)
res <- single.analysis(data$data, vp.pos=viewpoint, qWd = 2.5)
```

where important parameters are the *viewpoint*, which can be set at the centre point of the target restriction enzyme fragment, the *window*, which sets the region around the viewpoint which will be treated, and *qWd* with sets the stringency of the peak calling. A plot showing significant peaks can be generated with the command:

```
plot_C(res)
```
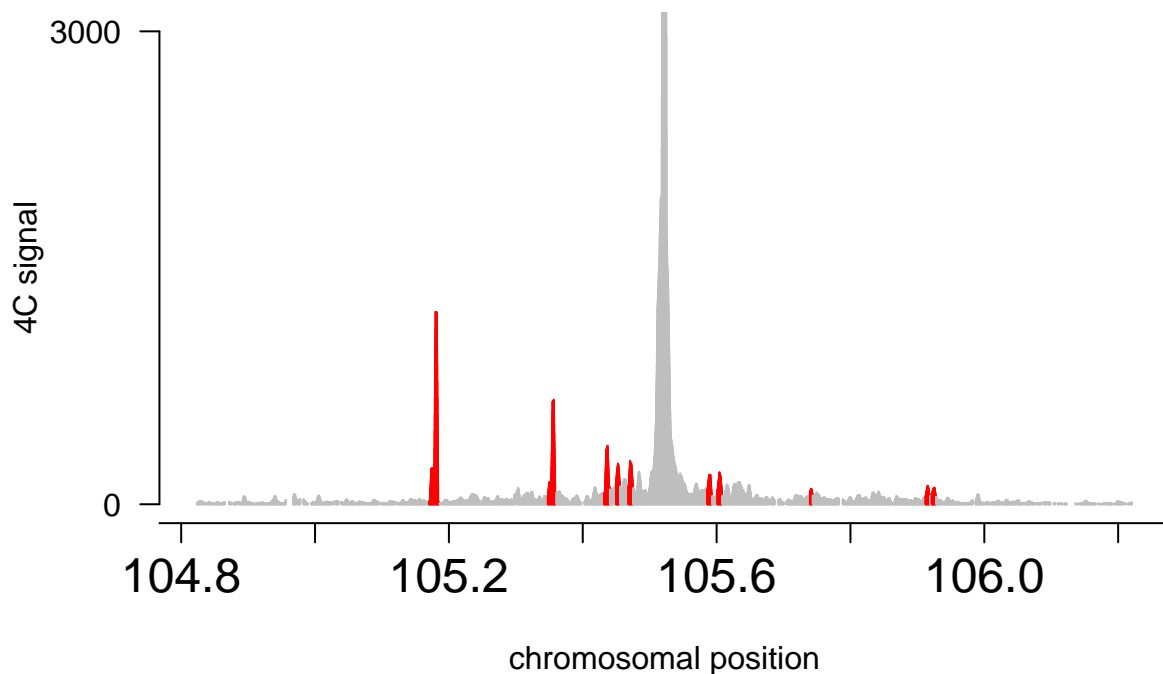
The peaks can then be accessed with

```
res$peak
```

The authors of the peakC software recommend that it is used with replicate data, as detailed in their documentation. We note that wig files generated from capC-MAP output using the script above should be read using the *readqWig* or *readMultipleWig* functions provided by peakC, and not the *readMultiColumnFile* function. For the *readMultipleWig* function, the required input is a list of wig files for the same target from different replica experiments (i.e. **not** a set of files for different targets).