**Code Usage** for Beagan et al. 2017

Before running any script, make sure you have NumPy, SciPy, R / Rscript, and multiprocess installed. pyBigWig and pandas will be needed to generate the pileups presented in figures 2B and 5FG. A virtual environment has been included that will allow the script usage – to activate it: “source virtual\_environment/bin/activate”

There are several pre-processing steps before plots are generated:

1. Start in the directory “Stage\_1\_PreProcessing\_for\_Figs\_3ACDEFHIJK\_4all\_5ABDEFGI\_6ABCEFGHI”
   1. This starts with raw counts files of sequencing reads mapped to the 5C primer set (these are found in ‘input’)
2. Run the bash script ‘preprocessing\_pipeline.sh’
   1. This takes the data through the quantile normalization and joint express matrix balancing stages
   2. There will be deprecation warnings but the code will run successfully.
3. Copy the final results ( 1 - the ‘primers\_augmented\_trimmed.bed’ file in the output directory; 2 - all of the .counts files in the ouput/expressed\_counts directory ) into the input directories of:
   1. Figs\_3ACDEK\_4AB\_5A\_6BCG
   2. Figs\_3FIK\_4B\_5B\_6EH
4. Starting in the ‘Figs\_3FIK\_4B\_5B\_6EH’ directory, run the bash script ‘No\_PP\_Removal\_Pipeline.sh’
   1. This bins the data into 4 kb genomic bins and calculates expected models. These datasets are then printed into counts files, found in output/counts.
   2. The scripts utilized are found in the ‘scripts’ sub-directory and the order they are utilized in is specified in the No\_PP\_Removal\_Pipeline.sh file.
   3. The counts files that are depicted in heatmap form are printed to their specified output sub-directories:
      1. Fig\_3AK\_4AB\_5A\_6BCG\_Sup\_Fig\_3
      2. Fig\_3C\_Sup\_Fig\_4
      3. Fig\_3D\_Sup\_Fig\_5
      4. Fig\_3E\_Sup\_Fig\_6
   4. The ‘zoom-in’ heatmaps depicted in Figs 3K, 4B, 5A and 6CG were plotted underlaid with ChIP-seq data; the accession numbers for the external datasets utilized are found in Supplemental Table 4.
5. Similarly, in the ‘Figs\_3FIK\_4B\_5B\_6EH’ directory, run the bash script ‘PP\_Removal\_Pipeline.sh’
   1. This step removes low confidence primer-primer pairs before binning the data into 4 kb genomic bins, calculating expected models, correcting the counts for the expected, fitting the data, calculating p-values and interaction scores, and classifying the data into loops.
   2. The scripts utilized are found in the ‘scripts’ sub-directory and the order they are utilized in is specified in the PP\_Removal\_Pipeline.sh file.
   3. The counts files that are depicted in heatmap form are printed to the specified output sub-directories ‘Fig\_3FK\_4B\_5B\_6EH’
      1. These ‘zoom-in’ heatmaps depicted were plotted underlaid with ChIP-seq data; the accession numbers for the external datasets utilized are found in Supplemental Table 4.
   4. The final list of looping clusters are printed to the output subdirectory ‘Fig\_3I’
6. The same can be repeated for the YY1 Knockdown 5C data, starting in ‘Stage\_1\_PreProcessing\_for\_Figs\_7C’ and proceeding to ‘Figs\_7C’.

The steps above have also generated intermediate files that are used to generate the remaining figure panels. These files can then be copied into the input sub-directories of the panel figure directories:

1. From ‘Figs\_3FIK\_4B\_5B\_6EH/output’ to ‘Fig\_3H\_Sup\_Fig\_8/input’:
   1. trimmed\_4kb\_pixelmap.bed
   2. compiled\_pvalues.txt
2. From ‘Figs\_3FIK\_4B\_5B\_6EH/output’ to ‘Fig\_3J/input’:
   1. trimmed\_4kb\_pixelmap.bed
   2. All counts files in the ‘Fig\_3J\_4E’ output subdirectory.
3. From ‘Figs\_3FIK\_4B\_5B\_6EH/output’ to ‘Fig\_4D\_5E\_6A\_Sup\_Fig\_12I-M/input’:
   1. trimmed\_4kb\_pixelmap.bed
   2. loops.json
   3. All counts files in the ‘Fig\_3J\_4E’ output subdirectory
4. From ‘Figs\_3FIK\_4B\_5B\_6EH/output’ to ‘Fig\_4E\_Sup\_Fig\_9/input’:
   1. trimmed\_4kb\_pixelmap.bed
   2. All counts files in the ‘Fig\_3J\_4E’ output subdirectory
5. From ‘Figs\_3FIK\_4B\_5B\_6EH/output’ to ‘Sup\_Fig\_7/input’:
   1. All counts files in the ‘counts’ subdirectory that end in ‘16kb\_4kb\_obs\_over\_max\_donut\_ll.counts’

Each of these figure panel subdirectories has a bash script at the highest level of the directory with the same name as the directory that contains the proper usage for all scripts within the directory needed to generate those figure panels. To simply regenerate the figure panels as they are, running the bash script with the command ‘bash script\_name’ will run the scripts properly.

Dependencies:

The following package versions were installed on the virtual environment in which these scripts were run:

BaseSpacePy==0.3

bctpy==0.5.0

biopython==1.66

bx-python==0.7.1

cutadapt==1.8.1

cycler==0.9.0

Cython==0.20.2

deepTools==2.4.3

dill==0.2.5

docutils==0.13.1

et-xmlfile==1.0.1

h5py==2.5.0

hifive==1.3.0

HTSeq===0.6.1p1

iced==0.2.2

interlap==0.2.3

ipython==1.0.0

jdcal==1.0

joblib==0.9.4

lib5c==0.4.1a9

libarchive-c==2.7

lockfile==0.12.2

MACS==1.4.3

MACS2==2.1.0.20150731

matplotlib==1.5.0

mirnylib==0.0.0

mpi4py==2.0.0

multiprocess==0.70.4

nose==1.3.0

numpy==1.10.1

numpydoc==0.6.0

openpyxl==2.3.0

palettable==2.1.1

pandas==0.17.0

patsy==0.3.0

Pillow==3.3.0

py2bit==0.2.1

pybedtools==0.7.9

pyBigWig==0.3.2

pycurl==7.19.5.1

pydevd==0.0.6

pyfaidx==0.4.8.1

pyparsing==2.0.6

pysam==0.8.4

python-dateutil==2.4.2

pytz==2015.7

PyX==0.12.1

rpy2==2.4.3

scikit-learn==0.17.1

scipy==0.18.1

seaborn==0.7.1

six==1.10.0

statsmodels==0.6.1

stevedore==0.10

tornado==3.1

virtualenv==15.1.0

virtualenv-clone==0.2.4

virtualenvwrapper==4.1.1

xlwt==1.0.0

**Pileups** (Fig\_2B\_Sup\_Fig\_2A\_2G and Fig\_5FG\_Sup\_Fig\_12A-F):

cycler==0.10.0

matplotlib==1.5.3

numpy==1.11.3

pandas==0.19.2

pyBigWig==0.3.2

pybedtools==0.7.9

pyparsing==2.1.10

pysam==0.10.0

python-dateutil==2.6.0

pytz==2016.10

six==1.10.0

wsgiref==0.1.2