**Note – you will have to edit paths to get most of the programs to work correctly.**

**Description of programs and datasets**

1. The analysis uses several datasets:

1. DNMsByStudy.txt - a compilation of all the DNMs used in this study. The spreadsheet gives the location of the DNM in bed format (i.e. chromosome, position and position+1), a code where the first letter corresponds to the dataset (K = Kong, N = Francioli, W = Wong, J = Jonsson, M = Michaelson), the reference and alternative alleles (i.e. the DNM), whether the mutation occurs at a CpG site and whether it is a transition or transversion.
2. The hg19 assembly of the human genome sequence.
3. Mutability indices from Michaelson et a. (2012). These are not provided but can be obtained from Jake Michaelson.
4. Divergence estimates along various lineages of the primate phylogeny, estimated using the method of Duret and Arndt. These are in the folder Divergence.
5. The average values for various genomic variables at the 100KB and 1MB scale. These are found in the Features folder.

2. Programs – compilation of datasets

1. summariseDNMs.py – adds the number of DNMs up in each of the 9 mutational categories for each of the 5 datasets and returns the counts and the number of sites.
2. numberOfSites.py – counts the number of sites in windows/blocks of the size specified and divides them into the categories of CpG C sites, non-CpG sites C sites and non-CpG T sites (CpG G sites are added to CpG C sites…etc). The program generates files with filenames of the form NumberOfSites\_1MB.txt, which gives the chromosome number, the block number (position of first base of block divided by the block size).
3. processWongCallableData.py – a program to count the number of callable trios per block/window as estimated for the study of Wong et al. This generates files of the form WongCallable\_10KB.txt. Because this program takes some time to run there is a second program which sums the data from the 10KB file to generate the files for 100KB, 1MB and 10MB.
4. sumWongCallable.py – does the summing of the 10KB data.
5. processDNMsByStudy.py – program to sum the number of DNMs in each block/window for each of the datasets. Combines this with the number of sites from the files NumberOPfSites\_scale.txt, WongCallable\_scale.txt, AggarwalaRates\_scale.txt and MichaelsonRates\_scale.txt to generate a file which lists chromosome number, block number, number of sites, number of DNMs in each dataset, the number of callable trios in Wong data, the predicted mutation rates from Aggarwala et al. and Michaelson et al.
6. processDNMsByStudyDetailed.py – program to process the DNM data from each dataset and generate a file with chrNo, blockNo, no. of CpG C sites, non-CpG C sites, non-CpG T sites, and DNMs in each of the 9 mutational categories. If the dataset is Wong then the number of callable trios in each of three categories is also given.
7. aggregateAggarwala.py – a program to take the rates predicted by Aggarwala et al. (see their supplementary table 7). The rate is given for each of 3 mutations from the middle nucleotide of each 7mer sequence and its reverse compliment. The program sums the mutation rates for the three mutational types: e.g. summing TTTATTT>TTTTTTT, TTTATTT>TTTGTTT and TTTATTT>TTTCTTT. The same mutation rate is given to a 7-mer and its reverse compliment. The file generated is AggarwalaMutationRatesAggregated.txt – this has two columns the 7-mer and its mutation rate.
8. aggarwalaBlock.py – calculates the average mutation rate predicted by the Aggarwala model for a window/block size of 100KB, giving the number of sites analysed and the total rate. Generates the file AggarwalaRates\_100KB.txt.
9. sumAggarwalaRates.py – takes the AggarwalaRates\_100KB.txt file and sums the data to generate AggarwalaRates\_1MB.txt
10. processMichaelson.py – takes the mutability indices given by Michaelson et al. and a map of hg19 and hg18 positions and calculates the sum and mean mutation rate for each 100KB block. Generates file MichaelsonRates\_100KB.txt
11. sumMichaelsonRates.py – takes the output from above for 100KB and calculates the 1MB rates. Generates file MichaelsonRates\_1MB.txt
12. combineFeatures.py – takes the DNM data for all datasets and combines it with the features data at various scales to generate files with the name FeaturesDNMs\_scale.txt. It is possible to split the data by paternal age.
13. combineFeaturesDetailed.py – combines the features data with the DNM data from a particular dataset, separating the mutations out into the 9 mutational categories
14. processDivergenceHumanLineage.py – a program to process the files generated by Peter Arndt’s program to calculate substitution rates along a phylogeny. The data are xml files which give the base composition and substitution rates for the 9 mutational categories. Note that the CpG rate is actually the sum of the transition rate C>T and the transition rate at CpGs.
15. processDivergenceAllLineages.py – calculates the overall divergence along each of the branches. Note this requires calculation of the CpG rate and then the base composition, including CpG sites. Generates the files allDivergence\_scale.txt
16. combineDNMDivergenceAllLineages.py – combines divergence data for each branch from allDivergence\_scale.txt with DNM data from each dataset from DNMsByStudy\_scale.txt.
17. combineDNMDiversityRR\_detailed.py – combine DNM data for each dataset split into 9 mutational categories from DNMs\_dataset\_scale.txt with SNP data from Diversity\_scale.txt.
18. simulateSequenceEvolution.py – simulate a neutral sequence subject to a particular pattern of mutation, defined by the 9 mutational categories. Rates come from the DNM data via the Mathematica notebook “GC content analysis.nb”

3. Programs – analysis

1. Infer distribution human DNMs.nb – Mathematica notebook to estimate the distribution of rates across the genome assuming the rates are gamma distributed. Routines are also included to assess the goodness-of –fit.
2. Expected correlations.nb – Mathematica notebook to calculate correlations and their expected values, where the expectations are derived by simulation.
3. GC content analysis.nb – Mathematica notebook to estimate the distribution of equilibrium GC contents. The notebook also includes a routine that ranks and groups windows by their current GC content and calculates the average mutation rate for all 9 categories of mutation for each group of windows. These can then be imported into simulateSequenceEvolution.py, which simulates sequence evolution according to the mutation rate estimates.
4. simulateSequenceEvolution.py - simulates sequence evolution according to the mutation rate estimates outputting the equilibrium GC content.