**File S1. Extended Methods**

*1000 Genomes Project Phase 3 Genetic variation Dataset*

The compressed genotype files (\*.vcf.gz) and tab-delimited index (\*.tbi) files for all autosomal chromosomes were downloaded from:

<http://ftp.1000genomes.ebi.ac.uk/vol1/ftp/release/20130502/>

We downloaded the human sample panel file that contains population information of each individual included in the genotype dataset from:

<http://ftp.1000genomes.ebi.ac.uk/vol1/ftp/release/20130502/>[integrated\_call\_samples\_v3.20130502.ALL.panel](http://ftp.1000genomes.ebi.ac.uk/vol1/ftp/release/20130502/integrated_call_samples_v3.20130502.ALL.panel)

The 2,504 individuals included in this dataset were divided into five major groups (i.e., ‘super-populations’) using information in the human sample panel file. We processed the above file to create three ‘population files’ for the three major populations of interest (Africans, Europeans, East Asians), in which each file contains the IDs of individuals belonging to a given population group. For subsequent analyses, we used genotype data of 1,668 individuals belonging to the three major populations of interest.

*FANTOM5 Enhancers Dataset*

We downloaded the BED files listing the entire set of FANTOM-predicted enhancers active in the 41 human tissues from:

<http://enhancer.binf.ku.dk/presets/>

All coordinates were converted from 0-based to 1-based before running the subsequent analyses. We also downloaded the Transcription Start Site (TSS)-enhancer associations dataset:

<http://enhancer.binf.ku.dk/presets/enhancer_tss_associations.bed>

*HapMap Phase II Genetic Map Data*

We downloaded the HapMap Phase II genetic map that had been lifted from build 35 to GRCh37 from:

<ftp://ftp-trace.ncbi.nih.gov/1000genomes/ftp/technical/working/20110106_recombination_hotspots/HapmapII_GRCh37_RecombinationHotspots.tar.gz>

We carried out a linear interpolation to annotate the 1000 Genomes Project Phase III variants that are not included in the HapMap Phase II project with the estimated recombination rates (cM/Mb).

*Calculation of Average Recombination Rates per Tissue*

To calculate the average recombination rate for each tissue, we took all the autosomal enhancers in a given tissue and for each enhancer, annotated the variants with the recombination rates for that site using the data from the preceding section (“HapMap Phase II Genetic Map Data”) and calculated the average recombination rate (cM/Mb) for the enhancer. We next took the average of the average values of recombination rates for all the enhancers in a given tissue to represent the recombination rate (cM/Mb) for that tissue. Finally, we further converted the average recombination rate per tissue to parameters *SLiM* can recognize (i.e., crossover per bp) by multiplying 10-8.

*VCFtools* (v.0.1.13)

*VCFtools* was used to calculate Weir & Cockerham’s *F*ST and to create new VCF files containing genotype data within each enhancer region in the FANTOM dataset. To calculate global *F*ST for each enhancer, we used the following command:

**./vcftools --gzvcf input.vcf.gz-- chr # --from-bp start\_pos --to-bp end\_pos --remove-indels --weir-fst-pop AFR\_samples\_list --weir-fst-pop EUR\_samples\_list --weir-fst-pop EAS\_samples\_list --out output**

‘input.vcf.gz’ file refers to the compressed genotype files, sorted by chromosomes, downloaded from the 1000 Genomes Project ftp site. ‘#’ indicates the chromosome id. ‘start\_pos’ and ‘end\_pos’ refer to the start and end positions of the enhancer, in base pairs. We used the weighted *F*ST values recorded in the \*.log file created from the above command. Next, we created new VCF files that contain the genotype data for each enhancer using the coordinates recorded in the FANTOM BED files:

**./vcf --gzvcf input.vcf.gz --chr # --from-bp start\_pos --to-bp end\_pos --remove-indels --recode --recode-INFO-all --out output**

This command creates a new VCF (output.recode.vcf) file with genotype data encompassing the regions defined by the ‘--from-bp’ and ‘--to-bp’ arguments. Finally, we created separate VCF files that contain the genotype data for the region that spans 50kb up and downstream of a given enhancer for each population as follows:

**./vcf --gzvcf input.vcf.gz --chr # --from-bp start\_pos\_50kb --to-bp end\_pos\_50kb --remove-indels --min-alleles 2 --max-alleles 2 --keep pop\_list --recode --recode-INFO-all --out output**

‘start\_pos\_50kb’ and ‘end\_pos\_50kb’ refer to positions 50kb upstream and downstream of the start and end positions of an enhancer as recorded by FANTOM, respectively. ‘pop\_list’ is the file that lists all the individuals belonging to a given population group. The ‘--min-alleles 2’ and ‘--max-alleles 2’ arguments will filter out multi-allelic sites, as *Selscan* only accepts bi-allelic sites as input data to calculate metrics of extended haplotype homozygosity. As a final step, we also manually filtered out duplicate entries for the same variant in the resulting VCF file. We refer to this file as ‘input\_final.vcf’ in the subsequent section (‘*Selscan* (version 1.2.0)’).

*Selscan* (version 1.2.0)

To calculate *nS*L values, we used the following command:

**./selscan --nsl --vcf input\_final.vcf --maf 0.01 --threads 16 --out output**

Apart from the minor allele frequency cutoff value (0.01), we used the default settings for other parameters. We used 16 threads to carry out calculations of *nS*L via *Selscan*, as shown above. We used the maximum of the *nS*L values (6th column) recorded in the \*.nsl.out file to represent each enhancer.

*Tabix* (version.0.2.6)

We used *Tabix* to compress the newly generated VCF files and create tab-delimited index (.tbi) files required for subsequent analyses via *PopGenome*. The commands used are as follows:

**./bgzip -c output.recode.vcf > output.vcf.gz**

**./tabix -p vcf output.vcf.gz**

*SLiM* (version 2.4.1) *and Simulation of Neutral Evolutions*

For each tissue, we used *SLiM* to simulate neutral evolution and generate VCF files of variants resulting from the action of neutral evolution alone, given past demographic events. We used *SLiM*’s implementation of Gravel et al’s pre-computed parameters of human demographic history for our neutral simulations. More specifically, at the beginning of the simulation (i.e. generation 1), the ancestral African effective population size was set to 7,310, which next expanded to 14,474 approximately 148,000 years ago (i.e. 5,920 generations ago). Approximately 51,000 years ago (i.e. 2,040 generations ago), the non-Africans split from Africans; the initial effective population size of these non-Africans was set to 1,861. The migration rates between Africans and non-Africans were set to 15 x 10-5. Next, approximately 23,000 years ago (i.e. 920 generations ago), the above-mentioned non-African population split into European and East Asian populations, with the initial effective population size for East Asians set to 554. In the same generation, the European effective population size was reduced to 1,032. The following migration rates were established for the remainder of the simulation: 2.5 x 10-5 for between Africans and Europeans, 0.78 x 10-5 for between Africans and Asians, and 3.11 x 10-5 for between Europeans and East Asians. Between generations 57,080 and 58,000, the European and East Asian populations were set to experience increase in their effective population sizes: for Europeans, the exponential coefficient was 0.0038 and 0.0048 for East Asians. A fixed mutation rate of 2.36 x 10-8 and recombination rate of 1 x 10-8 were used. For each tissue, we defined the length of the genomic element being simulated as the length of average of the lengths of all autosomal enhancers expressed in a given tissue. After 58,000 generations (i.e. end of the simulation), we sampled 661, 503, and 504 individuals from the “simulated” African, European, and East Asian population, respectively, to match the number of individuals included in the 1000 Genomes Project dataset. We used the following arguments to ensure that pairs of genomes being sampled belonged to the same individual:

**p1\_sample = p1.individuals;**

**sampled\_p1 = sample(p1\_sample, 661);**

**p2\_sample = p2.individuals;**

**sampled\_p2 = sample(p2\_sample, 503);**

**pe\_sample = p3.individuals;**

**sampled\_p3 = sample(p3\_sample, 504);**

p1, p2, and p3, correspond to the simulated African, European, and East Asian populations. Finally, to specify the output format as VCF, we used the arguments as shown below:

**sampled\_individuals = c(sampled\_p1, sampled\_p2, sampled\_p3);**

**sampled\_individuals.genomes.outputVCF();**

Genomic regions spanning 50kb upstream and downstream of the average enhancer length of a given tissue were created using the same approach as described above, except that the recombination rate for each tissue was set to the values obtained as described in a previous section (“Calculation of Average Recombination Rates per Tissue”). For each tissue, we generated 1) 10,000 “neutrally simulated” VCF files that contain genotype information for the “simulated” enhancer region and 2) an additional 2,500 “neutrally simulated” VCF files that contain genotype information for the “simulated” region that spans 50kb upstream and downstream of a “simulated” enhancer. Calculation of Tajima’s *D*, Weir & Cockerham’s *F*ST, *nS*L, and H12 using these VCF files were carried out as described in previous sections. We refer to these metrics as values generated under “neutral expectations”.

*GO Enrichment Analyses using TopGO (version 2.32.0)*

We first downloaded the GO annotations file (download date: May 17th, 2018) from:

<http://www.geneontology.org/page/download-go-annotations/goa_human.gaf.gz>

We processed the file downloaded from the above link and created a “Gene Universe (GU)” file for each tissue: this file would contain all the Gene Ontology (GO) IDs associated with any given putative target gene associated with an enhancer that is active in a given tissue. The GU file was created using only the autosomal putative target genes of enhancers in any given tissue. We also created “Target Genes (TG)” file for each tissue and each of the metrics we had calculated: more specifically, we obtained a list of GO IDs for the autosomal, putative target genes associated with enhancers that exhibit significantly extreme values of any given metric compared to those calculated on the neutrally simulated sequences. After further processing the files into a format required for *TopGO* analyses, we carried out the enrichment analyses via *TopGO* using the classic Fisher’s test (statistic = “fisher”) and the “weight” algorithm (algorithm = “weight”). This analysis was repeated for the three “root” GO terms (“Biological Process”; ontology = “BP”, “Cellular Component”; ontology = “CC”, “Molecular Function”; ontology = “MF”). We further carried out a FDR-based multiple test correction using R to obtain the final list of significantly enriched GO terms for each tissue of any given metric.

*Semantic Similarity Analyses using GOSemSim*

We used the results obtained from the preceding section (“GO Enrichment Analyses using TopGO”) to calculate pairwise semantic similarity of the GO terms between tissues for the *nS*L metrics calculated in all three populations: this analysis was carried out only for the *nS*L metric, as only this metric had more than one tissue with at least 5 significantly enriched GO terms. We used the “org.Hs.eg.db” Bioconductor package for the genome-wide annotation data of humans, as follows:

**SemData <- godata(‘org.Hs.eg.db’, ont = “BP”, computeIC = “FALSE”)**

We changed the argument “BP” to “CC” and “MF” to run the calculations for the three different root GO terms. ‘computeIC’ argument was set to “FALSE” as we were using a graph-based method for our calculations. We carried out pairwise semantic similarity calculations using Wang’s graph-based method and combined the similarity scores via the Best-Match Average (BMA) strategy as follows:

**semantic\_similarity <- mgoSim(first\_tissue\_data, second\_tissue\_data, SemData, measure = “Wang”, combine = “BMA”)**

‘first\_tissue\_data’ refers to a R object that contains the list of GO terms (obtained from the preceding section) for one tissue, while ‘second\_tissue\_data’ stores the list of GO terms for another tissue.

*Transposable Elements Analyses using BEDOPS (version 2.4.35)*

To obtain a genome-wide annotation of repeat regions, we downloaded the RepeatMasker (rmsk) track for the hg19 genome assembly (download date: June 15th, 2018) from UCSC genome browser:

<http://hgdownload.cse.ucsc.edu/goldenpath/hg19/database/rmsk.txt.gz>

We further converted this file into a BED format by grabbing the 6th, 7th, 8th, 12th, 11th, and 10th columns and carried out sorting using BEDOPS (“sort-bed”). We further filtered this file (“background\_regions.bed”) to include only those genomic regions that are annotated as the following: “LINE”, “SINE”, “LTR” (“background\_TE\_regions.bed”). We next took the enhancers that exhibit significant deviations from neutral expectations for each metric in each tissue and using the coordinates recorded in the original BED files, created respective BED files (“test\_regions.bed”). We next used BEDOPS to overlap the two types of BED files as follows:

**bedmap --echo --echo-map-id-uniq test\_regions.bed background\_TE\_regions.bed**

*GWAS catalog Data (v.1.0.2)*

The GWAS catalog was downloaded from (download date: June 2nd, 2018):

<https://www.ebi.ac.uk/gwas/docs/file-downloads/gwas_catalog_v1.0.2_associatons_final.gz>

For any given tissue, we obtained a list of the rs IDs of the SNPs that lie within the enhancers that displayed statistically extreme values of the calculated metrics compared to those calculated on the neutrally simulated sequences, and used those SNPs to query the GWAS catalog and obtain a list of significantly associated traits.

*Linkage Disequilibrium Calculations using Plink (v1.90b6.2)*

We first processed the VCF files for each chromosome (downloaded as described in “1000 Genomes Project Phase 3 Genotype Dataset” Section) as follows: 1) we first filtered out indels (i.e., included only SNPs) and 2) included only Europeans (n = 503). To obtain a list of SNPs that are in linkage disequilibrium (i.e., r2 = 1) with those used in the preceding section (GWAS catalog Data (v.1.0.2), we calculated Linkage Disequilibrium (LD) using Plink as follows:

**./plink --vcf genotype\_data.vcf --exclude duplicate\_snps\_list --ld-snp-list --r2 --inter-chr 1**

**--out output\_prefix**

“genotype\_data.vcf” refers to the filtered VCF file generated above and “duplicate\_snps\_list” is a file listing the duplicate rs IDs. We used the resulting SNPs’ rs IDs to query the GWAS catalog, as described in the preceding section.