SUPPLEMENTARY FILE M&M

Algorithms and programs

By processing the database of 1000 Genomes with our vrGVdatabase.pl program, a complete set of 16,326,219 vrGVs (named vrGVdatabase.txt) has been created. This file is available as Supplementary File “Data” (folder InputData). Following the 1000 Genomes data structure, each row of vrGVdatabase.txt represents one vrGV. The first 5 fields of vrGVdatabase.txt rows are identical to the corresponding fields from the VCF file of “1000 Genomes” characterizing these rare genetic variants. All individuals that have a particular vrGV, are also specified in the row corresponding to this vrGV in the vrGVdatabase.txt file. The information about these individuals is located in columns 6 to 9. For the simplicity of recognizing human individuals we combined their population identifiers with individual identifiers. For example, our combined identifier GBR_HG00255 refers to the individual HG00255 from the Great Britain population (GBR). A complete set of all vrGVs possessed by a particular individual may be obtained from vrGVdatabase.txt as an individual-specific table of vrGVs using Unix grep command. For example: “grep GBR_HG00255 vrGVdatabase.txt > GBR_HG00255”. In this command line example, the individual from Great Britain population (GBR) has an identifier HG00255 and his individual-specific vrGV dataset is saved into the file with the name matching his identifier (GBR_HG00255 filename). This computational procedure has been automatically repeated by Perl script get_vrGVdatabase.pl. It produced a complete set of 1092 individual-specific vrGV databases, which is available as Supplementary File “Data” (folder OutputDataWindow20).

The number of vrGVs among 1092 individuals varies from 9,245 to 84,450 in the files GBR_HG00155 and LWK_NA19373 respectively. An example of one such file (FIN_HG12345) is present in the Table 1. This figure exemplifies that shared vrGVs between pair of individuals are often clustered inside short genomic segments as described in our previous publication (Al-Khudhair, et al. 2015). We named these segments as shared Rare Variant Clusters (RVCs) and defined them according to the following rules: 1) a group of five or more vrGVs should be shared between a pair of individuals and these shared vrGVs should locate in close proximity to each other; 2) no more than four other non-shared vrGVs of these two individuals may be located inside the group of shared vrGVs. In order to characterize RVCs, a Perl program, named RVC.pl, have been created. This program scans an individual-specific vrGVs database (e.g. GBR_HG00255) and identifies all RVC regions that this particular person (in this example, GBR_HG00255) shares with any of the other 1091 individuals. RVC.pl produces three output files: GBR_HG00255_dat3, GBR_HG00255_dat4, and All_Pairs.txt files. The program uses a scanning window representing a specific number (window size parameter, $SW, default value 20) of consecutive rows of the individual-specific database of vrGVs.

When 5 or more shared vrGVs from any particular individual are counted within the scanning window they are selected as a new RVC, the starting position of which is marked by the first shared vrGV in the second column of the output “dat3” file (e.g. file GBR_HG00255_dat3) of the RVC.pl program. The program uses a stretching
window approach to continue counting number of shared vrGVs when sharing continues beyond the scanning window. This program counts the total number of shared vrGVs in the most stretched window and outputs this number in the column 10 of the "dat3" file (e.g. GBR_HG00255_dat3 file). The column 9 of this file represents the length of RVC in nucleotides (distance between the first and the last shared vrGVs). The personID_dat4 file is the derivative of the corresponding "dat3" file. When two particular individuals from "dat3" files share several RVC, these RVC regions will be combined and presented in a single line in the corresponding "dat4" file. For example, the line the CHB_NA18563_dat4 file:
“CHB_NA18563 CHS_HG00651 3 1565757 22”
means that CHB_NA18563 and CHS_HG00651 individuals share three RVCs with total number of vrGVs equal 22 and total length of three RVCs equal to 1,565,757 nucleotides. The All_Pairs.txt file is the exact copy of "dat4" file when RVC.pl is executed for the first time. For multiple execution of RVC.pl program the All_Pairs.txt file accumulates (concatenates) all "dat4" files. The Perl script TotalStat_vrGVs_v3.pl runs automatically RVC.pl program on all 1092 input files representing all individuals. After such execution of TotalStat_vrGVs_v3.pl, it generates the summing output file All_Pairs.txt, which contains the exhaustive list of all pairs of individuals that share RVC genomic segments. (NOTE: prior to use TotalStat_vrGVs_v3.pl program, the All_Pairs.txt file should be empty). The All_Pairs.txt file (obtained after execution of TotalStat_vrGVs_v3.pl) was processed further with Table_Mix.pl program, which sorts the pairs of individuals sharing RVC(s) on the 14 sets of population-specific files: Table_ASW, ... Table_YR (when individuals from a pair belong to the same population), and also, Table_Mix file, representing all pairs composed of individuals from different populations. Then, Table_Mix file was used as an input for Inter_pop_results2.pl program, which sorts all inter-population pairs by the populations the individuals belong to. Specifically Inter_pop_results2.pl program creates 91 output files: result2_ASW_CEU, result2_YRI_TSI and so on. Each filename contains two population identifiers. So, all inter-population pairs from ASW and CEU that share RVC are present in the result2_ASW_CEU file. All inter-population pairs in these files share one or few RVCs. However, since close relatives may be present in the same population, the intra-population pairs may share dozens of RVC, and, thus, compromise statistical analysis of RVC distributions. In order to remove close relatives from intra-population pairs the Intra_pop_results2.pl program was executed 14 times for each population. This programs removes all pairs that share >=50 RVCs (default value for close relatives) and keeps all pairs that share less than 50 RVCs. It produced fourteen result2_ASW ... result2_YRI files. To obtain medians and averages of the shared RVCs for inter- and intra-population pairs, the median.pl program was executed. The results obtained with median.pl are present in the Table 4. To obtain distribution of RVC lengths, the files info_ASW_CEU ... info_YRI_YRI and length_ASW_CEU ... length_YRI_YRI were created using program statistics_length_RVC.pl.

Finally, the 1092x1092 table of shared RVC lengths for each possible pair of individuals has been produced by the Perl program TableRVClength.pl. It creates the output table Table_RVC_length. This program takes as input the entire set of
“result2...” files (result2_ASW, result2_ASW_CEU, etc.) and processes them pair by pair. The modification of this program -- TableRVCnumber.pl produces similar 1092x1092 table of RVC numbers per each pair. These Table_RVC_length and Table_RVC_number tables were processed by Microsoft Excel to make them colorful and convert them to heat-maps. These excel tables have been colored based on the number or length of shared RVCs and presented as heat-map figures (Figure 1A and 1B). The tables Table_RVC_length and Table_RVC_number are provided as Tables S1 and S2 in Supplementary File “Data” (folder OutputDataWindow20).

Important parameters for RVC_v2.pl program and matching probabilities for vrGVs

Initial parameter for the size of scanning window was 9 rows ($SW=9), which had been chosen empirically by viewing hundreds of data files for individuals from different populations and observing that very frequently shared vrGVs for a pair of individuals are immediately adjacent to each other. The initial stretching window step was also short (parameter $strW = 2) based on the same empirical observations. The results for this set of parameters are shown in Supplementary File “Data” (folder OutputDataWindow9). However, further in-detail analysis demonstrated that often this particular threshold ($SW=9) and small steps for stretching window ($strW=2) created a disruption of a long IBD segment into several IBD segments. For example, declared close relatives from 1000 Genome Project appeared to share several hundreds of IBD segments instead of no more than 40 according to the modeling in Figure 2. Further variation of parameters ($SW and $strW) demonstrated that reduction of number of IBD segments between close relatives close to the theoretically expected quantities could be achieved by increasing scanning window to $SW=20 and the stretching window step to $strW =40. The probability of random matching of a group of vrGVs between individuals increases with the rising of window size. It could be estimated by the formula:

\[ C_n^k p^k (1 - p)^{n-k} \]

where \( k \) is the number of shared vrGVs, \( n \) is the size of the window, and \( p \) is the frequency of vrGVs (we consider the highest chance of \( p=0.002 \)). The \( C_n^k \) takes into consideration all possible combinations of \( k \) vrGVs within the window of \( n \) rows. For our default parameters (\( k=5 \) and \( n=20 \)) the probability is \( 0.5 \times 10^{-9} \), which practically never happens by chance. These parameters ($SW=20 and $strW=40) have been chosen as a default ones and all the data presented in tables and figures are calculated with this default set.

Modeling of IBD segment inheritance

We created Perl program IBDsimulator.pl for modeling distribution of IBD chromosomal segment sizes that propagate in consecutive generations. This program takes into account the real values of meiotic recombination rates across human autosomes, which were obtained from the International HapMap Consortium genetic maps (Frazer, et al. 2007) available from ftp://ftp.ncbi.nlm.nih.gov/hapmap/recombination/latest/rates/genetic_map_chr1_b36.txt. The program assumes no selection preferences for the inheritance of genomic
segments. It models all autosomes and does not consider X and Y chromosomes. The program creates maternal and paternal autosomes for a model person, named primogenitor (generation 0, G0), and then computes the inheritance of the segments of these chromosomes in a chain of primogenitor’s descendants in multiple successive generations until all the primogenitor’s IBD chromosomal fragments are lost. Specifically, \textit{IBDsimulator.pl} creates a gamete from the primogenitor’s maternal and paternal chromosomes and transmit it to one offspring (first generation, G1). During the next computational cycle, the gametes of this offspring are computationally simulated using the data of the primogenitor’s chromosomes. The coordinates for these progenitor’s IBD segments are randomly chosen in accordance to the real distribution of recombination sites (HapMap tables). The details are provided in the comments to \textit{IBDsimulator.pl} Perl script available from Supplementary File “DATA”. A random gamete from the G1 person creates one of the parental chromosomes of one offspring in the next generation (second generation, G2). A random gamete from the G2 person is computed next, which creates one of the parental chromosomes of one offspring in the next generation (third generation, G3). This process continues for multiple downstream generations of the primogenitor. The program repeats these cycles in multiple successive generations while recording the IBD coordinates of the initial primogenitor genomic materials in every generation until all IBD segments from the primogenitor are lost. Using this program we examined IBD inheritance pattern in multiple descending generations for 500,000 independent primogenitors. The results are presented in the Figures 2 and 3 and in Supplementary File “Data” (folder Modeling).