Journal of
Proteomics & Bioinformatics

Research Article Open Access

GDF: Dealing with High-throughput Genotyping Multiplatform Data for Medical and Population Genetic Applications

Jorge Amigo[1]*, Antonio Salas[2], Javier Costas[3] and Ángel Carracedo[1,2,3]

[1]Grupo de Medicina Xenómica, CIBERER, Universidade de Santiago de Compostela, Galicia, Spain
[2]Unidade de Xenética Forense, Instituto de Medicina Legal, Facultade de Medicina, Universidade de Santiago de Compostela, Galicia, Spain
[3]Fundación Pública Galega de Medicina Xenómica (FPGMX), Hospital Clínico Universitario. 15706, Santiago de Compostela, Spain

Abstract

Background: A number of different high throughput genotyping platforms have arisen recently. These platforms generate large amounts of genotyping data which is subsequently processed and stored in public and/or private databases. Both, the variety of platforms employed by the different laboratories and the large amount of data they generate, entail serious problems for data managing in most laboratories. Some public or private software packages available today solve some important needs, but they deal with the data from a point of view that the researcher may probably not share, and no supervision of the results (e.g. genotyping inconsistencies or summaries of the genotyping data) may be performed.

Results: The main goal of the Genotyping Data Filter (GDF) software is to allow the researcher to locally manage large numbers of genotypes generated by the most standard genotyping platforms, obtaining statistics and summaries of the genotyping experiments whilst maintaining their privacy. GDF also allows the user to supervise the data such that the researcher can easily evaluate important parameters, including the proportion of missing data in samples and single nucleotide polymorphisms (SNPs), Hardy-Weinberg equilibrium, etc. Additionally, GDF parses the raw data into different text formats needed as input files in popular software packages frequently used in medical and population genetic applications.

Conclusions: GDF is a Perl program that efficiently process data from various genotyping platforms, allowing researchers to easily inspect their own genotyping data and to parse it for a wide spectrum of well-known specialized analysis software. It has been prepared to be run through a user friendly web interface on the most common cases, but it can also be run as a local script on personal computers, or even supercomputers for very large-scale projects.

Background

A major interest of current genomics research is devoted to disease-gene association studies, that is, studies aimed to identify DNA variants presumably associated with susceptibility (or protection) to a common disease. Advances in genotyping and sequencing technologies, coupled with the development of sophisticated statistical methods, have afforded investigators novel opportunities to define the role of sequence variation in the development of common human diseases. [1,2]. Considering that during the last years the genotyping efficiency has heavily increased, research groups have now to cope with genomic large-scale and high density SNP association analysis through all the genome. It is expected that these genome-wide association studies may identify alleles related to complex disorders, and therefore finding the underlying causative relationships is currently a major challenge.

It is estimated that SNPs occur once per 100~300 bases in the human genome, which represents over 10 million SNPs in our whole genome [3]. Thus, in large-scale association studies, genotyping all SNPs in a candidate region for a large number of individuals is still costly and time-consuming. Sets of nearby SNPs on the same chromosome are inherited in blocks (this pattern of inherited SNP variants on a single block is a haplotype), and although blocks may contain a large number of SNPs and can be very variable in size, only few SNPs might be needed to uniquely tag and identify the haplotypes in a block (what is called a haplotype tagging SNP, or htSNP or tagSNP). This is due to the correlation between alleles at nearby variant sites, named linkage disequilibrium (LD), that exists because of the shared ancestry of contemporary chromosomes that is erode by mutation and recombination [4]. From the initial efforts to characterize the human genome by studying its common variability [5,6], the HapMap Project was born as a public effort to build a map of these haplotype blocks and their htSNPs. This map of blocks and htSNPs allows reducing significantly the number of SNPs required to interrogate the entire genome for association with a disease phenotype from more than 14 million SNPs that exist today to roughly 500,000 htSNPs. This will make genome scan approaches to find regions that affect diseases in a much more efficient and comprehensive way, since effort will not be wasted typing more SNPs than necessary and all regions of the genome can be included. Results from these whole genome scans promise to be successfully translated into useful applications in areas such as medical diagnosis [7] or pharmacogenomics [8]. HapMap is then a useful resource that allows selecting a group of SNPs to analyze a possible association between custom genomic regions with the studied pathology. HapMap, together with other ambitious genomic projects (e.g. Perlegen), has allowed changing the classical perspective of analyzing a single functional polymorphism on a single gene to the current analysis of multiple genes from the same pathway, or even the whole genome.

Several and very different genotyping techniques have arisen in

***Corresponding author:** Jorge Amigo, Grupo de Medicina Xenómica, CIBERER, Universidade de Santiago de Compostela, Galicia, Spain. E-mail: jorge.amigo@usc.es

Received October 22, 2011; **Accepted** December 27, 2011; **Published** January 02, 2012

Citation: Amigo J, Salas A, Costas J, Carracedo Á (2012) GDF: Dealing with High-throughput Genotyping Multiplatform Data for Medical and Population Genetic Applications. J Proteomics Bioinform 5: 001-006. doi:10.4172/jpb.1000206

less than a decade. Companies like Sequenom [9], Applied Biosystems [10,11], Illumina [12] or Affymetrix [13,14] have been developing their own exclusive techniques to identify SNPs with very diverse throughput capacities (which is constantly increasing due to continuous innovations, most of them in chemistry) but also developing different strategies in terms of hardware and software. What they all have in common is that they can be used for large-scale genotyping experiments, and so they all have to face the same issue: data management. The software packages that the companies usually provide with their genotyping platforms have been developed with the main aim in mind of making the generated data management as comfortable and powerful as possible. However, the lack of flexibility and serious limitations of these software packages encourages for local dedicated developments. In addition, it is often required to use multiple genotyping platforms to perform a single experiment, as the genotyping methods are very different and certain SNPs may be better detected with one or other genotyping technique. Therefore, corporate software does not usually allow dealing with all the experiment data as a whole. The complexity of these tools will well vary with the specific needs of each group: from a simple set of platform-specific Visual Basic macros like TIMS [15] to SNPator [16], an example of an *ad hoc* online package designed to cover the needs of the large-scale genotyping process of the Spanish National Genotyping Centre (http://www.cegen.org/) on multiple platforms.

The high throughput genotyping (HTG) capability does not only depend on the genotyping techniques, but also on the data handling approaches that had to manage all that new overwhelming amount of information [17]. The critical issues that arise on HTG projects always concern the data: inspecting it for possible errors, the whole management and the later analysis. As mentioned above, some useful free tools have appeared during the HTG expansion in order to cover the management part using databases and web interfaces [18,19], even with great visual aids [20], while they implement internal consistency checks and embed different algorithms for data analysis. Data managers such as SNPator [14], SNPP [17] or SNPLims [16] are also capable of exporting the data in appropriate formats for later deeper analysis, a basic request for any HTG project as it is very hard to implement all the algorithms that all users may need, but the limited format offer for some researchers may still force them to find a more appropriate tool such as GDF, that provides the input format for several different programs on the association and population studies field.

Although SNPator's data importing module implements many of the features considered in GDF, a major advantage of GDF is that it can work locally, and this feature may be of great help for researchers that have to deal with low to medium SNP genotyping projects, especially for those researchers that wish to preserve as much as possible the privacy of their research projects. Although web-based implementations are obviously useful, some researchers may not be fully comfortable with the idea of storing their data in servers where they do not have full control of it, and this may even lead to some bioethical concerns. GDF represents a much more flexible alternative that could even be embedded into a larger software package already developed, or into any local pipeline for specific research needs.

Implementation

The GDF has been designed to work as a flexible interface between the researcher and the raw genotyping data dumped by the platforms (Figure 1). The researcher may need to process the raw data in a particular way, and for that reason the proprietary software from the genotyping platforms may be not very flexible. The main idea is to allow merging complementary information with the raw data, allowing the researcher to obtain not only general summarizing reports, but also to perform a customizable quality control of the results and to have that raw data parsed into input files for specific analysis software packages (Table 1).

Data reading module

The reading process of the raw data exported directly from the genotyping platforms works line by line, as each line contains a single genotype. Most platforms share this characteristic in plain text tabulated files, and therefore, by using a different recognition pattern for each platform, it is possible to identify and dissect even any forthcoming technology.

Each platform has its own format, but all of them provide at least the information concerning the codes of the SNPs that were genotyped, the sample IDs and the genotyping calls. The reading module of GDF

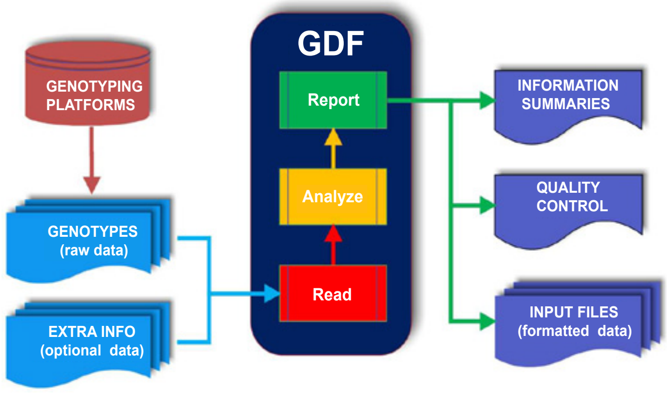

Figure 1: The GDF data workflow. The raw data coming from the genotyping platforms is directly read by GDF, while extra information may be added to characterize it. This combination is then processed and exported if requested into input files for specific software of analysis, along with a complete report of all the internal checks performed.

INPUT	Genotyping Platforms	Sequenom SNPlex Illumina Affymetrix*
	Generic formats	HapMap SNPs vs. Samples tables Samples vs. SNPs tables
OUTPUT	Association Studies	GeneHunter Haploview` PLINK MEGA-2 Arlequin Phase Hapblock EMLD Unphased MDR
	Population Studies	Structure Arlequin Haploview

Table 1: File formats handled by GDF. Being a flexible interface between raw data and specific analysis programs, GDF deals with several input formats coming directly from the genotyping platforms, from the HapMap project, or even from custom made tables. The Affymetrix format coverage is not full though (*), as only general reports may be obtained from it through GDF due to its usual data size, but for the rest of the input formats GDF is able to parse their genotypes and provide the input for several programs of analysis for association and population studies.

also allows to store additional information collected by the different platforms (plate information, well position on the plate, manual edition of the call or the score of the result for instance). All this information may be later analysed and used to enrich the output, but only these three fields are mandatory.

Once the line is recognized by the pattern recognition engine the reading function starts to process it, saving the platform detected and storing all the dissected data into an appropriate hash indexed by gene, sample ID and SNP position. The first and third indexes are automatically given by GDF unless they are indicated in a complementary input file, but they must be always consider to allow the user to group sets of SNPs and to sort them in case this is needed.

Complementary input files

Since the raw data from the genotyping platforms can possibly not contain all the data the researcher may need, GDF gives the option of providing extra information through complementary input files, and this information is then concatenated with the raw data. The kind of information that can be supplemented in these extra files could be related to the need of grouping information (e.g. by ethnic or population groups), sample characterization, translation for non-explicit platform alleles, or even signalling samples that should be excluded from some particular analysis.

Configuration file

This is a three column tabulated plain text file where the SNPs can be grouped in genes or any kind of grouping strategy. A number (e.g. chromosome location) can also be attached to each SNP. The first row must contain the fields' names, which must be "GENE", "SNP_ID" and "POSITION", and the rest of the lines must contain the data.

This file is often used to divide the output in different files, one for each gene, to sort the SNPs by their position to build the appropriate haplotypes or to filter SNPs to be processed as GDF will not process any SNP that is not present in this file.

Pedigree and population file

Pedigree and population information may be assigned to each processed sample respectively in pedigree format and in a tabulated text headed file with three columns: samples, populations and population ids. This may be desired when expecting GDF to provide in its output the input formats for specific programs, such as Structure [21] (will not work without population information) or Phase [22] (the case-control running option will not be available if the appropriate column is not present in the input file) for instance. The GDF web interface detects which files have been uploaded, and allows the user to select only the available outputs for the data that is to be used.

Allele translation file

Raw data coming from platforms such as Taqman, Illumina or even older versions of the SNPlex format do not provide explicit calls for each genotype. A code is given to each result instead of the appropriate base (al1 or al2, A or B, A1 or A2 …), and the translation of that code is SNP specific and then stored in a configuration file. To deal with that code in the output data from any of the mentioned platforms GDF allows importing a tabulated text file that, without any headers, indicates in three columns the SNP name, the code and its translation. This information is then used to convert internally all the data to the proper formats and to give the desired outputs.

SNPs and samples not to be processed

As the raw genotyped data file may be difficult to edit, and being this not the best option, a filtering option was implemented through file input. Thus, if a list of SNPs or samples is provided (text files with all the desired SNPs or samples in a single column) GDF will not process the data associated with them. This may help, for instance, to remove from the statistics or from the specific output files samples or SNPs that have been wrongly genotyped due to experiment errors, or to treat separately data coming from different projects that have shared the experiment but that should not be analysed together.

Results

We have developed a program written in Perl, as it is one of the most popular reference programming language for fast and comprehensive text handling [23]. GDF performs a series of internal analysis such as quality control and consistency checks, and provides formatted data to be given as input for several different programs for association and population studies (Table 1).

Several projects present in the literature have used GDF as the data pre-processing tool when they had to deal directly with raw genotypes (e.g. [24,25]). As a practical example, GDF was used to deal with 137.015 genotypes generated by the Sequenom platform, creating the input files for additional analysis using Structure, Unphased, Haploview, PHASE, and MDR, in 31 seconds. This led to the replication of *DTNBP1* as a schizophrenia susceptibility locus [26].

Another common use of GDF is to parse tables of samples and SNPs that may have been manually edited, or data from the HapMap project extracted directly from its website or from intermediate repositories such as SPSmart [27,28]. For instance, the study of the *CYP21A2* gene reveals a low SNP density on HapMap, but it can be merged through GDF with the SNP information obtained by direct sequencing of 21-hydroxylase deficiency patients [29] in order to highlight haplotypes associated with this pathology.

Internal Data Analysis

Validation analyses tests

This group of analysis includes all the error and consistency checks performed by GDF. One of the primary aims of this sub-program is to let the user revise the raw data from the platform, and thus it allows the user to know if there are any incoherencies (e.g. a duplicated genotype does not match) present for a given genotype. If control samples were introduced in the experiment, or any sample is just genotyped more than once, a quality control will be carried out in order to detect inconsistencies.

Another analysis performed is the check for more than two alleles found for a single SNP. Considering that bi-allelic SNPs are the most common variation, highlighting these situations is needed as they would normally represent a flagrant error, possibly at the experimental design of the experiment or at the genotyping software assignation.

Informative analyses

All the rest of the tests performed by GDF are meant to describe the data analysed, in order to provide a broader understanding of the experiment. The most important ones would be the detection of data skews (inconsistent genotypes for the same SNP tested on the same sample), monomorphic SNPs, and the highlighting of SNPs or samples with no valid result in all the experiment, but the summary and statistics performed with GDF given at the end of each run are also very informative: the total of the data input lines is presented, and compared with the total numbers of genotypes detected, valid results and failed genotypes. Table 2 gives a detailed description of these checks. In case repeated genotypes (such as quality controls or just replicas) are entered, GDF will display a quality control section at the output, detailing the numbers of repeats, how many did actually match and how many did not match.

Output Files

Files under demand

These files contain all the available input formats for the specific analysis programs. Currently, the linkage pre-makefile format is included, which is valid for popular association studies programs such as GeneHunter [30], Haploview [31] or PLINK [32], or for a meta-analysis software package named MEGA-2 [33] that is able to support 28 target programs by combining the linkage format with a pedigree and a mapping file. Additionally, GDF can also be used to obtain the input format for other popular association studies software such as Arlequin [34], EMLD [35], Hapblock [36], MDR [37], Phase [20] and Unphased [38,39]. Input formats for population studies software like Structure [21] may also be obtained. Except the input format for Arlequin, the rest of the programs will work directly with the files generated by GDF. Arlequin needs some minor manual edition in order to include the configuration headers.

Automatically generated files

This set of files contains valuable information obtained from the input files. For instance, the GDF generates a set of files that contain information on genotyping errors or undesired inconsistencies. There are also three useful files that are generated by default: i) a text table containing a matrix of SNPs versus samples which will state all the results in an efficient manner for visual inspection, ii) a statistic file for samples with information of the percentage of missing genotypes on each one, and iii) a statistics file containing information about the alleles observed, SNP heterozygosity values, minor allele frequencies, or the result of checking for Hardy-Weinberg equilibrium and its statistical significance using a simple chi-square test.

Performance in memory and time

As GDF is meant to work with large amounts of data coming from HTG experiments, its performance had to be measured in order to predict the running time and the computer resources that could be needed for the biggest experiments. For that reason we tested its performance on an ordinary PC with an Intel Pentium IV 3.4GHz processor and 1GB RAM. We then measured the computational resources demanded by GDF with respect to the number of genotypes that had to be processed. The summary of the benchmarking results of GDF are reflected on the top graph of Figure 2, where the memory and time linear tendencies can be observed, validating the adequacy of the internal GDF code which could otherwise depend exponentially on the amount of processed data. This fact is critical to allow future evolution of the program.

Unused genes	Genes present in the configuration file with all their SNPs untested
Unknown SNPs	SNPs present in the data file but not in the configuration file
Untested SNPs	SNPs present in the configuration file but not in the data file
Failed SNPs	SNPs that failed in all the genotyped samples
Failed samples	Samples with no successful genotype
No pedigree samples	Samples with no pedigree information present in the pedigree file if used
Unperformed tests samples	SNPs that were not genotyped on a sample but they were tested on the rest
Overlapping information samples	Samples that carry a third allele, assuming most genotyping techniques deal with bi-allelic SNPs

Table 2: Analyses performed by GDF. A description of the internal checks that GDF performs in order to improve the description of the data being analysed, which are given at the end of each run.

The lack of perfect linearity regarding the demand of memory and time needed by GDF as more SNPs and samples are processed seems to indicate that there are underlying factors affecting GDF's performance (see the bottom graphs of Figure 2), such as the percentage of repeated genotypes present on the experiment (quality control) that significantly reduces the amount of memory needed. The invested time per genotype strongly depends on the platform used, as there are platforms that provide much more information for each genotype that is also processed by GDF. Providing more information implies more complex line pattern recognition, and that is the major latency present on the program. Thus, platforms like SNPlex include in their outputs information concerning genotyping quality scores and manual edition flags.

Interacting with the program

GDF can be run locally in command line, executing its code through a locally installed Perl interpreter. This is the most versatile option, and as there are plenty of versions of Perl depending on the operating system, GDF has been designed in order to be also independent to the platform. In addition, as some researchers may not be comfortable with command line commands, several graphical user interfaces (GUIs) have also been developed to work around this issue: i) an online PHP interface to the most updated version of GDF, which runs it directly on the web server without having to install anything locally, and ii) a Visual Basic interface that runs an encapsulated executable version of GDF for Windows platforms only. In both cases the user gets a four steps interface: i) the data input, where all the files that are going to be used must be selected, ii) the options selection, where all the GDF's options may be chosen, iii) the formats request, where the programs to which the data should be formatted for should be highlighted, and iv) the final results. In this last step there will always be a screen output, accompanied by a link to all the files that were generated (one of them will be that screen output for later inspection).

Discussion

Performing of all of the automatic analyses described above provides more information about the raw data than the one provided by corporate software, independently from the genotyping platform used. The personalization of the analyses performed allows GDF users to find out data which is difficult to retrieve when using those corporate software packages, because of the appropriate options absence or the manual revising impossibility, such as platform errors or even pre-genotyping problems.

The genotyped data is never the final step of the full analysis process. The data must always be processed by deeper analysis and specialized programs that will go far beyond to find information such as associations (case-control, TDT,…), haplotypes, present population substructures, and so on. Some platforms may give as output the input for a certain analysis program, but the researcher may prefer to be able to transform any kind of data coming from any platform into any

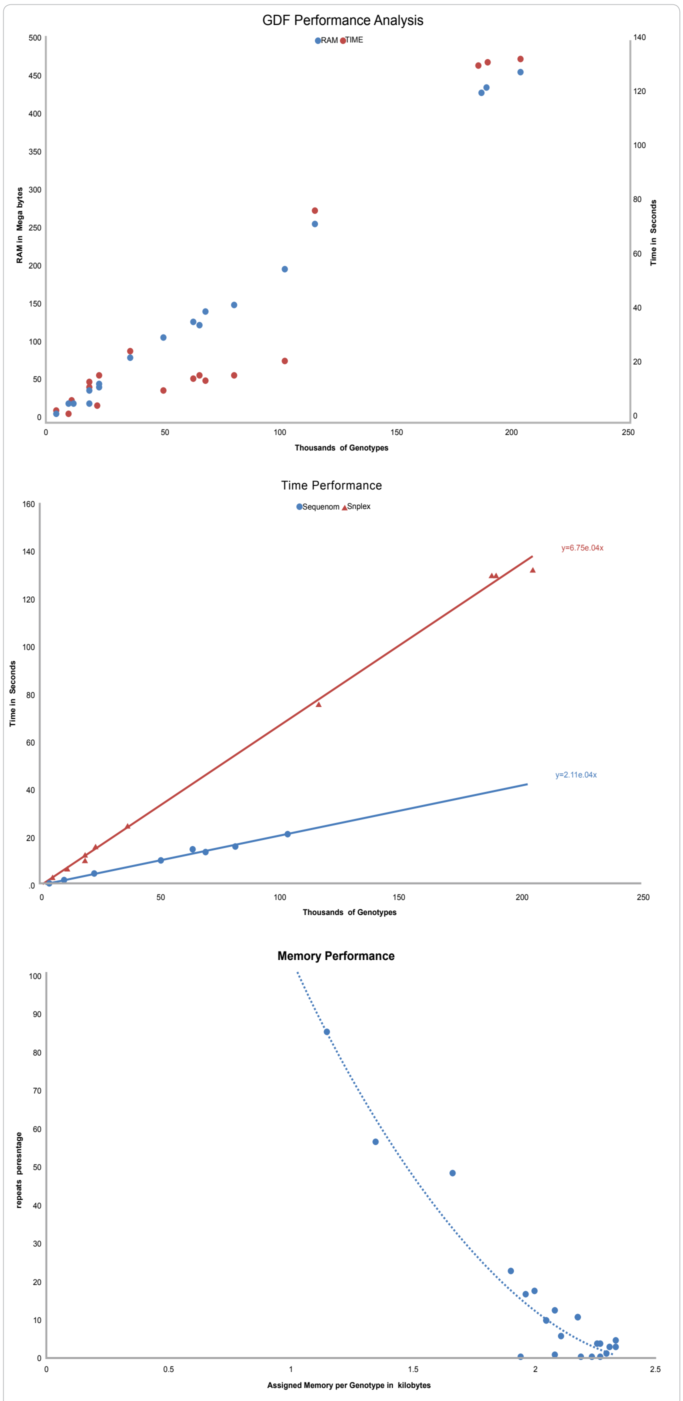

Figure 2: Performance analyses. Multiple experiments performed with Sequenom and SNPlex were grouped and summarized for this figure. Raw measurements of memory and time are presented on 2A to show the linear dependency of the resources needed, and a deeper analysis of those magnitudes is displayed on the two bottom graphs. The performance in time showed on 2B highly depends on the platform where the genotyping data comes from, due to the complexity of the pattern recognition applied to each line. The memory assigned to each genotype in each experiment is presented on 2C, depending on the amount of information that each genotype may be characterized with. This assigned memory does not really depend on the genotyping platform, but it drastically depends on the percentage of genotype repeats of the experiment.

desired format. GDF allows doing this through an appropriate internal variable structure and design that is prepared to easily deal with new upcoming input formats.

Providing a parallelizable version of the program is our next aim, as it would allow running it either on dedicated supercomputers or directly on personal computers with multiple-core machines.

Conclusion

GDF is a program to process HTG data specially produced by the biomedical community. Other fields of research are now benefiting from HTG such as those interested in quantitative characters' analysis in species of commercial interest, for instance. Any researcher may then workaround some of the corporate software packages' limitations embedding GDF in the genotyping routine. A set of improvements to this process have been implemented inside GDF, and their use is fairly straightforward. But the best advantage for the researcher is probably not to be forced to use a local database, which will need expertise on installation and maintenance, nor even a remote one that could compromise the data privacy. Previous similar work has been done using a SNP database to hold the data while processing it, but GDF allows dealing directly with the data.

Authors' Contributions

JA carried out the design, programming and implementation of the software, and drafted the manuscript. AS, JC and AC participated in the design of the software, suggesting and testing the implementation of new features, and helped to draft the manuscript. All authors read and approved the final manuscript.

Acknowledgements

This work was supported by the Genoma España foundation through the National Genotyping Center (CeGen), and by grants from the Xunta de Galicia (Grupos Emerxentes; 2008/037), Ministerio de Ciencia e Innovación (SAF2008-02971), and Fundación de Investigación Médica Mutua Madrileña (2008/CL444) given to AS. Thanks to Beatriz Sobrino, María Torres and Inés Quintela for their feedback on their usage as genotyping platform managers, to Ceres Fernández and Laura Fachal for their analysis on additional usage possibilities, and to the many users that have provided us with the feedback about features to improve.

References

1. Cordell HJ, Clayton DG (2005) Genetic association studies. Lancet 366: 1121-1131.
2. Crawford DC, Nickerson DA (2005) Definition and clinical importance of haplotypes. Annu Rev Med 56: 303-320.
3. The International HapMap Consortium (2003) The International HapMap Project. Nature 426: 789-796.
4. Altshuler D, Brooks LD, Chakravarti A, Collins FS, Daly MJ, et al. (2005) A haplotype map of the human genome. Nature 437: 1299-1320.
5. Johnson GC, Esposito L, Barratt BJ, Smith AN, Heward J, et al. (2001) Haplotype tagging for the identification of common disease genes. Nat Genet 29: 233-237.
6. Sachidanandam R, Weissman D, Schmidt SC, Kakol JM, Stein LD, et al. (2001) A map of human genome sequence variation containing 1.42 million single nucleotide polymorphisms. Nature 409: 928-933.
7. Ribas G, Gonzalez-Neira A, Salas A, Milne RL, Vega A, et al. (2006) Evaluating HapMap SNP data transferability in a large-scale genotyping project involving 175 cancer-associated genes. Hum Genet 118: 669-679.
8. Deloukas P, Bentley D (2004) The HapMap project and its application to genetic studies of drug response. Pharmacogenomics J 4: 88-90.
9. Jurinke C, Oeth P, van den Boom D (2004) MALDI-TOF mass spectrometry: a versatile tool for high-performance DNA analysis. Mol Biotechnol 26: 147-164.
10. De la Vega FM, Lazaruk KD, Rhodes MD, Wenz MH (2005) Assessment of two flexible and compatible SNP genotyping platforms: TaqMan SNP Genotyping Assays and the SNPlex Genotyping System. Mutat Res 573,111-135.
11. Tobler AR, Short S, Andersen MR, Paner TM, Briggs JC,et al. (2005) The SNPlex genotyping system: a flexible and scalable platform for SNP genotyping. J Biomol Tech 16: 398-406.

12. Steemers FJ, Gunderson KL (2005) Illumina Inc Pharmacogenomics 6: 777-782.

13. Hardenbol P, Yu F, Belmont J, Mackenzie J, Bruckner C, et al. (2005) Highly multiplexed molecular inversion probe genotyping: over 10,000 targeted SNPs genotyped in a single tube assay. Genome Res 15: 269-275.

14. Moorhead M, Hardenbol P, Siddiqui F, Falkowski M, Bruckner C, et al. (2006) Optimal genotype determination in highly multiplexed SNP data. Eur J Hum Genet 14: 207-215.

15. Monnier S, Cox DG, Albion T, Canzian F (2005) T.I.M.S: TaqMan Information Management System, tools to organize data flow in a genotyping laboratory. BMC Bioinformatics 6: 246.

16. Morcillo-Suarez C, Alegre J, Sangros R, Gazave E, de Cid R, et al. (2008) SNP analysis to results (SNPator): a web-based environment oriented to statistical genomics analyses upon SNP data. Bioinformatics 24: 1643-1644.

17. Li JL, Deng H, Lai DB, Xu F, Chen J, et al. (2001) Toward high-throughput genotyping: dynamic and automatic software for manipulating large-scale genotype data using fluorescently labeled dinucleotide markers. Genome Res 11: 1304-1314.

18. Orro A, Guffanti G, Salvi E, Macciardi F, Milanesi L (2008) SNPLims: a data management system for genome wide association studies. BMC Bioinformatics 9: S13.

19. Zhao LJ, Li MX, Guo YF, Xu FH, Li JL, et al. (2005) SNPP: automating large-scale SNP genotype data management. Bioinformatics 21: 266-268.

20. Tebbutt SJ, Opushnyev IV, Tripp BW, Kassamali AM, Alexander WL, et al. (2005) SNP Chart: an integrated platform for visualization and interpretation of microarray genotyping data. Bioinformatics 21: 124-127.

21. Pritchard JK, Stephens M, Donnelly P (2000) Inference of population structure using multilocus genotype data. Genetics 155 :945-959.

22. Stephens M, Smith NJ, Donnelly P (2001) A new statistical method for haplotype reconstruction from population data. Am J Hum Genet 68: 978-989.

23. Stein LD (2001) Using Perl to facilitate biological analysis. Methods Biochem Anal 43: 413-449.

24. Salas A, Vega A, Milne R, García-Magariños M, Ruibal A, et al. (2008) The 'Pokemon' (ZBTB7) Gene: No Evidence of Association with Sporadic Breast Cancer. Clin Med Oncol 2:357-362.

25. Vega A, Salas A, Milne RL, Carracedo B, Ribas G, et al. (2009) Evaluating new candidate SNPs as low penetrance risk factors in sporadic breast cancer: a two-stage Spanish case-control study. Gynecol Oncol 112: 210-214.

26. Vilella E, Costas J, Sanjuan J, Guitart M, De Diego Y, et al. (2008) Association of schizophrenia with DTNBP1 but not with DAO, DAOA, NRG1 and RGS4 nor their genetic interaction. J Psychiatr Res 42: 278-288.

27. Amigo J, Phillips C, Salas A, Carracedo A (2009) Viability of in-house datamarting approaches for population genetics analysis of SNP genotypes. BMC Bioinformatics 10: S5.

28. Amigo J, Salas A, Phillips C, Carracedo A (2008) SPSmart: adapting population based SNP genotype databases for fast and comprehensive web access. BMC Bioinformatics 9: 428.

29. Loidi L, Quinteiro C, Parajes S, Barreiro J, Leston DG, et al. (2006) High variability in CYP21A2 mutated alleles in Spanish 21-hydroxylase deficiency patients, six novel mutations and a founder effect. Clin Endocrinol (Oxf) 64: 330-336.

30. Li H, Schaid DJ (1997) GENEHUNTER: application to analysis of bipolar pedigrees and some extensions. Genet Epidemiol 14: 659-663.

31. Barrett JC, Fry B, Maller J, Daly MJ (2005) Haploview: analysis and visualization of LD and haplotype maps. Bioinformatics 21: 263-265.

32. Purcell S, Neale B, Todd-Brown K, Thomas L, Ferreira MA, et al. (2007) PLINK: a tool set for whole-genome association and population-based linkage analyses. Am J Hum Genet 81: 559-575.

33. Mukhopadhyay N, Almasy L, Schroeder M, Mulvihill WP, Weeks DE (2005) Mega2: data-handling for facilitating genetic linkage and association analyses. Bioinformatics 21: 2556-2557.

34. Excoffier L, Laval G, Schneider S (2005) Arlequin (version 3.0): an integrated software package for population genetics data analysis. Evol Bioinform Online 1: 47-50.

35. Huang Q (2005) EMLD.

36. Zhang K, Qin Z, Chen T, Liu JS, Waterman MS, et al. (2005) HapBlock: haplotype block partitioning and tag SNP selection software using a set of dynamic programming algorithms. Bioinformatics 21: 131-134.

37. Hahn LW, Ritchie MD, Moore JH (2003) Multifactor dimensionality reduction software for detecting gene-gene and gene-environment interactions. Bioinformatics 19: 376-382.

38. Dudbridge F (2003) Pedigree disequilibrium tests for multilocus haplotypes. Genet Epidemiol 25: 115-121.

39. Dudbridge F (2008) Likelihood-based association analysis for nuclear families and unrelated subjects with missing genotype data. Hum Hered 66: 87-98.