Computational candidate gene prioritisation by genomic data fusion

Stein Aerts[*1,2#], Bert Coessens[1#], Diether Lambrechts[3], Robert Vlietinck[4,5], Yves Moreau[1], Bart De Moor[1]

[1]Department of Electrical Engineering (ESAT-SCD), University of Leuven, Kasteelpark Arenberg 10, 3001 Leuven, Belgium

[2]Present adress: Laboratory for Neurogenetics, Flanders Interuniversity Institute for Biotechnology, University of Leuven, Belgium

[3]The Center for Transgene Technology and Gene Therapy, Flanders Interuniversity Institute for Biotechnology, University of Leuven, Belgium

[4]Department of Human Genetics, Flanders Interuniversity Institute for Biotechnology, University of Leuven, Belgium

[5]Department of Population Genetics,Genomics and Bio-informatics, University of Maastricht, the Netherlands

[#]These authors contributed equally to this work.

Email: Stein Aerts[*]- stein.aerts@esat.kuleuven.ac.be;

[*]Corresponding author

Abstract

We present a method for the prioritisation of any list of candidate 'test' genes, based on the combination of multiple information sources, and given a set of related 'training' genes representing a biological case. This general approach allows us in human genetics to identify disease genes, but also in molecular biology to rank and identify members of biological pathways or genes with a particular function or localisation. The prioritisation is done by comparing summarised information of the test genes with that of the training genes. The information sources that are currently consulted are textual descriptions from MEDLINE abstracts and LocusLink records, Gene Ontology annotations, Interpro protein domains, KEGG pathways, protein interactions from BIND, EST-based expression data, microarray expression data from multiple experiments, sequence similarities, transcription factor binding sites and *cis*-regulatory modules.

We describe the methods of the applied summarisation and of the order statistics that are used to combine the prioritisations according to all individual information sources. Our approach is validated by a cross-validation on 29 training sets from the Online Mendelial Inheritance in Man (OMIM) database and several test sets of random genes. We further assess the contributions of the individual data types to the overall gene ranking, calculate their pairwise dependencies, and assess the bias to known genes. Lastly, we describe the prioritisation of the 3p22-25 chromosomal region and of the whole chromosome 3 to find the best candidate genes to be putatively involved in congenital heart defects, cardiomyopathies, and arrhytmias. In conclusion, we propose a fast way to prioritise large lists of genes based on multiple information sources, combine them, and rank them in a biologically meaningful way.

The additional file contains, for each of the three heart diseases (arrythmias, cardiomyopathies, and congenital heart defects) the tables with the training genes, short summarisations of each trained submodel, the top 10 scoring test genes from region 3p22-25, a selection of the top 50 scoring genes of chromosome 3, and specificity measurements.

Introduction

The high-throughput post-genomic technologies used nowadays in molecular biology, often fail to deliver immediate biological clues that lead to an improved understanding of biological systems. Instead, researchers are often confronted with large lists of candidate genes or proteins that require further filtering and validation in the lab. Some examples of research approaches where this can be the case are (1) microarray and proteomics experiments that compare two conditions generating lists of differentially expressed genes; (2) microarray and proteomics experiments with multiple conditions (time-series analyses for instance) generating clusters of co-expressed genes; (3) *cis*-regulatory sequence analyses that involve genome-wide searches of putative target genes (1; 2; 3).

Likewise in the field of genetics, linkage mapping and association studies often result in relatively large chromosomal regions containing tens to hundreds of genes as putative candidate disease genes (4). The positional candidate gene approach, that has been quite successful in pinning down the gene that is responsible for the linkage, uses information of the function, homology, and expression pattern of all the genes in the mapped region, to select a few of the most promising candidates to validate further. This validation is often done by expression studies in model organisms or fine mapping and association studies. The manual prioritisation process, to choose which genes to validate, mostly lies in the hands of the expert and depends mostly on the combination of several lines of biological evidence and publications. Acknowledging that the multitude of biological information cannot be captured in the mind of a single biologist, there has been some recent effort in automating or at least assisting in the process of gene prioritisation using computational methods. Given that multiple, sometimes contradictory sources of information need to be reconciled and that their information must be balanced, we borrow the term "data fusion" from engineering to characterise this process (the term "data fusion" is itself derived from "sensor fusion" where it characterises the integration of the information from multiple sensors in a robotic device).

Three categories of methods can be recognized. The first group can be described as *ab initio* methods that predict the disease (or process) susceptibility of a gene, or that prioritise genes based on location (does it lie within a region of linkage), sequence, sequence phylogeny, annotation, or expression of the gene, or combinations thereof. Hauser *et al.* combined a

statistical measure of differential expression of a gene (measured with SAGE) between diseased and control individuals with location (linkage) data to select the best candidate genes for a particular disease, as illustrated for Parkinson's disease, and called this procedure genomic convergence (5). Franke *et al.* combined linkage and association maps with microarray expression data in a tool called TEAM (6). Turner *et al.* prioritized candidate disease genes based on the statistical over-representation of Gene Ontology and Interpro annotations. Lastly, López-Bigas and Ouzounis based their measure of disease probability purely on a gene's sequence and its evolutionary trace (7).

A second category of methods can be defined as pure classification methods, based on logistic regression, discriminant analysis, artificial neural networks, support vector machines, etc. To our knowledge these have not been applied to classify disease genes yet, most probably because of the small sizes of most training sets, and because of the difficulty of defining negative training samples (i.e., genes excluded to be involved in the process).

A third category could be defined as prioritisation methods that measure the *similarity* of a gene with a disease, or the similarity with genes causing the disease, and prioritise a gene list based on this similarity. As opposed to the first category of methods, they rely on the existing knowledge of a disease. In this case, small training sets can be used, and negative samples are not necessarily needed. The first systematic study of gene prioritisation with purely computational techniques was done by Perez-Iratxeta *et al.* (8). In a text-mining approach they used MEDLINE abstracts, MeSH vocabularies, and Gene Ontology annotations to prioritise genes based on the similarity of these descriptions to those of a disease. Freudenberg and Propping (9) measured the similarity of a gene with disease genes of related diseases using GO annotation data. A more basic example of similarity matching is implemented by van Driel *et al.*(20) who integrated location data, expression data, and phenotypic information in the GeneSeeker web application that allows to filter genes based on the similarity with, or rather the co-occurrence of, user-defined query terms.

The method we propose in this work is a member of the third category. It is based on the assumption that a new candidate disease gene has properties similar to a set of other genes, for which the most obvious choice is the set of all disease susceptibility genes that are currently known for this disease. However, if none or only a few genes are known to be involved in a certain disease, genes involved in phenotypically related diseases could also be used, as is done in (9). The properties that we currently use are: Gene Ontology (GO) annotation, textual descriptions

from MEDLINE abstracts, microarray gene expression data, EST-based anatomical expression data, KEGG pathway membership, protein domain annotation of Interpro, *cis*-regulatory elements, and BIND protein interaction data. The similarity between the properties of two genes is measured either using correlation measures (for vector-based data) or using meta-analysis (for attribute-based data). The meta-analysis also takes genome-wide occurrences of attributes into account. All candidate genes are ranked according to the similarities of each property separately. The resulting rankings (e.g., ten) are combined into an overall rank and p-value using order statistics. Given that the information from numerous genomic sources are hedged to prioritise the candidate genes, we call this prioritisation genomic data fusion.

The proposed approach is generic and is not restricted to the prioritisation of genes according to their disease susceptibility. It can be used to perform any kind of prioritisation based on the similarity with any (coherent) gene group (for instance, the members of a biological pathway). However, the validation of the method described in this work uses disease susceptibility prioritisation as a working model.

Results

Computational gene prioritisation

In this paragraph we present our approach towards prioritisation of candidate genes with respect to a set of training genes using multiple heterogeneous information sources. Figure 1 overviews the different steps in the analysis.

In the first step a training set `TRAIN` is compiled and all information from the different sources described in the Data and methods Section is gathered and prepared. In the case of Gene Ontology annotations, KEGG pathway membership, EST-based expression data, and InterPro protein domains, the statistically over-represented attributes are determined (see the 'Meta-analysis' section in Data and Methods). For the textual information and the microarray gene expression data, the average profile of all individual gene profiles is taken. The BIND data are stored separately for each gene in `TRAIN`. The transcription factor binding site information of all training genes is compiled into one large vector. Also, the best combination of three transcription factors within human-mouse conserved non-coding sequences in the upstream sequences of the genes is recorded. For the sequence similarity a local BLAST database is created consisting of all coding sequences of the genes in `TRAIN`. We call all gathered data from one information source a submodel for this source. All submodels together form a model for `TRAIN`.

In the second step a set of candidate genes `TEST` is compiled (for instance, candidate disease genes or candidate pathway members). All genes in `TEST` are then scored against our model for `TRAIN`. For each test gene the necessary information is retrieved and used to calculate a similarity by comparing it to the information contained in the submodels. This results in a list of scores for each submodel.

In the third step the lists of scores are used to rank the genes in `TEST`. We then combine all rankings of the different submodels to obtain an overall ranking of the candidate genes by applying the order-statistics formula (see Data and Methods).

Large-scale cross validation

We have assembled 29 gene sets representing 29 human diseases from the OMIM gene_map table. Each gene set contains at least 9 genes, although HUGO-to-Ensembl mapping reduced the number of genes for some diseases. The smallest gene group was the one for Amyotrophic Lateral

Sclerosis (ALS) with only 4 Ensembl genes, and the largest one was the leukemia gene set with 113 genes. In total we had 627 disease associated genes with an Ensembl identifier and a disease gene set contained on average 19 genes.

Figure 2 describes the different steps in our cross-validation procedure. For each gene set, a leave-one-out cross-validation is performed: at each run, one gene is left out and the remaining genes are used as training set. Each of the available data types (see Data and Methods) is used to train a submodel, summing to 10 submodels in total. Then, the left-out gene and N_{rand} random genes (9, 49, or 99) are used as test set. For each size we constructed 100 random sets, out of which we randomly selected one set for each tested left-out gene. The rankings for each separate submodel, as well as the combined ranking for all submodels based on the order statistics are recorded.

Performance of the combined model

Figure 3 shows the Rank ROC curves for the rankings of all leave-one-out cross validations of the 29 OMIM diseases using 9, 49, and 99 random genes plus the left-out gene as test genes. The Rank ROC curves are based on the overall rank obtained by order statistics (see Data and Methods). In the same figure, the Rank ROC curve of the same leave-one-out cross validation using random training sets is also plotted. Our validation experiment results in a biologically meaningful prioritisation that is significantly better than random prioritisations. Generally, for such sizes of test sets, the prioritisation causes the left-out (disease) gene to be ranked among the top 50% of the test genes in more than 90% of the cases. In about 65% of the cases, the left-out gene is found among the top 10% of the test genes.

Performance of individual submodels

Figure 4.A shows the Rank ROC curves of all submodels individually, using test sets of 100 genes (99 random genes plus a left-out gene). Some data types perform well in the prioritisation, especially the Text, GO, and EST models. At first sight, the Rank ROC curves for the other seven submodels have a seemingly worse performance. However, the AUC of a cross-validation combining some of these submodels (KEGG, BIND, Motif, CRM, InterPro) is 77.1% (plot not

shown). In other words, the performance of the combined ranking can still be significantly better than random, even if only submodels with bad individual performances are used. It is therefore useful to include all models in the prioritisation, as we will do in our case study below. There can be several reasons why a submodel can have a low AUC in this large scale study, while it may still contribute to the combined ranking. One reason is that the AUCs are averages across 29 OMIM diseases of which some may be modelled well by certain submodels but less well by others. This can be illustrated by the fact that a cross-validation on Alzheimer's disease alone yields an AUC of 76.3% for the Atlas microarray data, much higher than the average AUC of all diseases. A second reason might be the high number of missing values for certain submodels. In the case of KEGG, for instance, this causes the curve to level towards a very low maximum sensitivity (equal to the number of left-out genes with KEGG information divided by the total number of left-out genes). A third cause is the way certain submodels are ranked. For the BIND submodel for example, all genes always have data, but prioritisation typically results in a very high ranking for a hit and the lowest ranking for a non-hit. Again the curve levels towards a low sensitivity because only a few genes are ranked high and none are ranked intermediate. The curve differs from that of the KEGG submodel in that it jumps to a sensitivity of 1 when (1-specificity) reaches its maximum (i.e., when the cutoff is set to the maximum rank). Nonetheless, if we compare the performance of these submodels with their performance if random training sets are used (see Figure 4.B), the AUCs are much higher for the former.

Pairwise dependencies between submodels

If some of the submodel rankings were correlated, then the combined ranking would be biased because the order statistics require independent rank ratios. In that case, nonetheless, the combined ranking could still be used as an approximate ranking if the obtained p-values are not considered in the decision of a cut-off (instead, the threshold could then be purely based on selecting a certain percentage of test genes). To test whether certain submodels are correlated, we recorded all individual rankings of the different submodels of 29 different sets of 50 random genes prioritised with the same 29 OMIM disease models as above. All pairwise Spearman correlations between these rankings are shown in Figure 5. All submodels are positively correlated with the overall order statistics rank ('Ra' in the figure). In other words, they all contribute to the overall

ranking. Those with a low correlation have a lot of missing data (for instance in the case of KEGG ('Ke') only 10.1% of the genes had data; in the case of the Atlas expression data ('At') only 34.4% of the genes had data). The pairwise correlations between the different models are all around zero, so we do not expect large biases in the order statistics rank. This means the p-values of the order statistics can be used as a significance threshold. The correlation between the InterPro and Blast submodels is slightly greater than all others (namely 0.18), which can be expected because protein domain similarities between test and training genes naturally makes good sequence similarity scores more likely.

Bias towards known genes

In general "there already *is* a bias in candidate gene selection towards well-characterised genes" (6). We expected that our approach should at least partly alleviate this bias, and should allow for unknown or less known genes to be ranked highly, because (1) genes are prioritised based on *multiple* information sources instead of one or a few; and (2) not only functional information (for instance, GO, text, pathways) is used, but also data sources that are equally valid for unknown genes (i.e., prior-knowledge-independent data), namely microarray gene expression data, EST-based gene expression data, protein domain predictions from InterPro, protein interactions, sequence similarities, and *cis*-regulatory data.

To establish the magnitude of the bias towards well-characterised genes we first investigated the influence of the number of information models for which a certain gene has data available, on the possibility for this gene to get a good rank. Figure 6.A shows for each number of available information models the percentage of genes that is ranked between 0 and 10, 10 and 20, and so on in a list of 50 random test genes that are prioritised according to our 29 disease models. We observe only a slight trend of higher rankings for genes with more available information. A second indicator of how well a gene is characterised can be the presence of a HUGO gene symbol. Therefore we plotted the same information in Figure 6.B for genes with and without HUGO gene symbol. Again only a slight bias can be seen towards known genes. It is apparent that even genes with very little information (from three submodels onwards), can be ranked in the top 10 in a test set of 50 genes.

A case study on disease gene hunting

To demonstrate the effectiveness of the prioritisation methodology, we performed a case study on a large chromosomal region. After reviewing the literature, chromosomal region 3p22-25 seemed appropriate for the following reasons: (1) it is a region of intense genetic activity to which several life-threatening abnormalities affecting formation or function of the heart have been mapped; (2) for each of these cardiac abnormalities, a representative set of training genes, causing similar disease phenotypes when mutated in humans, is available and (3) multiple genes within this chromosomal region are mutated in unrelated human disorders and may serve as a control to illustrate the specificity of the prioritisation process. The cardiac abnormalities previously mapped to the 3p22-25 locus are: an atrioventricular septal defect, which is a birth defect arising from insufficient atrioventricular septation during heart development (10); a dilated cardiomyopathy, which is a disease from the heart muscle associated with heart dysfunction (11) and a Brugada syndrome, which causes sudden cardiac arrest due to ventricular arrhythmia (12; 13).

We generated three sets of 10 to 15 genes mutated respectively in congenital heart defects (CHDs), cardiomyopathies (CMs) and cardiac arrhythmias (ARs; Table S1a-S3a). These sets of genes served as training sets for the prioritisation of putative disease genes within the 3p22-25 genetic locus. Consultation of the training results revealed which attributes were over-represented in the training set as compared to the full genome (Table S1b-S3b). Notably, each of the different submodels contributed to the overall picture, which was generated for each training set. For example, T-box or Zn-finger transcription factor domains (InterPro submodel) were hallmarks of the CHD genes, which were additionally typically involved in transcription, development of the heart, and DNA binding according to the Text submodel. Myogenin (MYO) and myocyte enhancer factor 2 (MEF2) transcription complexes (*cis*-regulatory module submodel) were present within the regulatory sequences of CM genes, for which the most significantly over-represented GO terms were: cell motility, (striated) muscle development and contraction, kinesin complex, cytoskeleton, myofibril and sarcomere. Each of the trainings set genes were also characterized by a high expression in the heart (EST expression submodel). Certain sequence motifs, such as the binding sites for nuclear-factor-of-activated-T-cells (NFATs) and MEF2 transcription factors, which are involved in cardiomyocyte differentiation and cardiac hypertrophy, were

over-represented within the AR and CM training sets (Motif submodel). Overall, the summarized information for the three training sets was in close agreement with the current understanding of the genetic pathways involved in their diseases. This indicates that the data summarisations are able to extract biologically relevant information for a set of disease-causing genes.

We then ranked the 210 test genes from chromosomal region 3p22-25, based on their similarity with one of the three training sets (Table S1c-S3c). To show that a prioritisation of a large number of test genes is also feasible, the same procedures were repeated for a test set consisting of all 1048 genes from Chromosome 3 (Table S1e-S3e). While scoring, we did not use the text-mining submodel because we based our assessement of the prioritisation mostly on literature searches. Note however that the text-mining submodel is one of the best performing models, and its exclusion from our case study is conservative since this submodel is able to infer similarity from non-trivial textual links as well, and not only from direct gene-disease references.

Upon inspection of the prioritised list, we found that highly ranked test genes were very similar to those within the training set (Tables S1c-S3c and S1e-S3e). Some of these genes were already known to cause human cardiac diseases—for this reason they had also been selected in the training set—(e.g., CRELD1 mutations cause CHDs, whereas mutations in cardiac channel SCN5A cause arrhythmias). Both genes were assigned with the highest rank in the CHD and RY list respectively. After noticing the highly ranked position of ACVR2 and SHOX2, we additionally discovered that mutations in these genes cause heterotaxy and Turner syndrome, which are both syndromatic disorders frequently associated with CHDs. Other highly ranked genes give rise to an identical disease phenotype when mutated in mice—for example, deficiency of LAMR1 and CAV3 causes cardiomyopathy in mice. This is a particularly striking observation, as all the used information, except perhaps that from the GO submodel, originates from experiments performed in a human experimental setting. In some instances, genes received a high rank because they scored very high in one of the submodels—for example, because they belong to the same protein family or contain the same InterPro domain as the training genes. SCN10A and SCN5A belong to the same family of sodium channels (the type alpha subunit sodium channel (ENSF00000000129)), whereas EOMES, TBX-1 and TBX-5 are all transcription factors characterized by the presence of a T-box transcription factor domain (IPR001699). Also genes homologous to one of the training genes, such as semaphorine 3B (SEMA3B), which is homologous to the SEMA3E training gene of the CHD set, received a high ranking in the BLAST

submodel. Remarkable is the fact that plexin-A1, a receptor for many of the semaphorine family members, also received a high rank in the CHD Chromosome 3 list. A recent study even reported that plexin-A1 is essential for cardiac morphogenesis during chick development. Many of the high-ranking CHD genes, such as WNT5a and WNT7A were involved in neural crest guidance. Defects in the migration of neural crest cells, which orchestrate the complex remodeling processes of the cardiac outflow tract, are well known to lead to a typical spectrum of cardiovascular and facial abnormalities. So we were able to select genes interacting with one of the training genes or identify genes involved in the same biological process.

In a few instances, however, it was not clear why certain genes received such a high position (e.g., CAPN7 and MYD88). This does not necessarily imply that these genes are false negatives. Interestingly, among the high-ranking genes, there were also some genes for which there was only an Ensemble Gene ID, but no gene name or any other reference in the MEDLINE citation database available. This indicates that, despite the presence of a slight bias towards known genes, the methodology can also be used for the discovery of novel putative disease-causing genes.

As a negative control, we calculated the average ranking position of all the other disease-causing genes in the 3p22-25 region, not yet included in any of the three training sets (n=10 for the 3p22-25 region). We also determined the average position of the top 10 ranked genes from the three training lists (Table 1Sd-3Sd; for example the average position of the top 10 genes from the CHD list was calculated in the gene list prioritized for the cardiomyopathic training set) and we calculated the e-value associated with these average positions. The values of the negative control sets were never significantly similar to the training set, which indicates that the prioritisations generated by our methodology are specific for each training set, and therefore also specific for each disease.

Software availability

We are developing a software tool called ENDEAVOUR that implements the described methodology in a fully automated way. A preliminary version together with a user manual can be downloaded from `http://www.esat.kuleuven.ac.be/endeavour`. All described submodels can be automatically trained on a user-defined training set. Disease- or process-specific microarray data—from in-house experiments or downloaded from public repositories—can be added as

submodels by the user. More detailed software functionalities will be described in a companion paper (B. Coessens et al., in preparation).

Discussion

Given the growing number of publicly available, post-genomic databases with information about human genes and proteins, we show that it is possible to integrate all these data to prioritise a set of genes based on how similar they are to a group of other genes, which we call training genes. Such a prioritisation is particularly useful for gene hunting in complex human diseases, in which case all known disease genes can be used as training genes. We have chosen a simple strategy that allows any number of heterogeneous data types to be integrated and weighted equally. For a set of training genes, any attribute-based gene annotation (for instance, Gene Ontology terms) is transformed into statistical over-representation values, as compared to their genomic occurrences. Vector-based gene annotations (for instance, microarray expression profiles) are averaged within a set. Certain genes under study, like positional candidate disease genes, are ranked according to these transformed annotations, and all individual rankings are combined into an overall ranking using order statistics.

The cross-validation yielded AUCs of around 85%. Such encouraging high numbers can stimulate the use of this kind of prioritisation methods in a laboratory environment. In case all test genes would eventually be validated in the lab, starting from the top scoring ones and until the hit is found, then the prioritisation does not confer any risk. It only increases the chance of finding the hit faster, thereby reducing required resources. In comparison with prioritisations done by experts, the computational prioritisation is orders of magnitude faster (e.g., one hour versus several weeks), and much more information can be added to the analysis. Another advantage, which is problably as important as the previous, is the statistical nature of the result, with applied corrections for multiple testing. Far too often, the identification of (disease-causing) genes remains based on a serendipitious fishing expedition in a large pool of candidate genes, whereby each of the genes is assessed as a true disease gene, tested against a wide variety of data sources and fitted into a multitude of hypothetical mechanisms underlying the disease. Besides its time-consuming aspect, this manual prioritisation method soon becomes complex and chaotic, frequently resulting in genes which become selected based on minimal biological evidence, without taking into account the multiple testings performed. For the first time, the method presented here offers a solution to the multiple testings performed in gene prioritisation procedures as it ranks genes based upon biological similarity to a training set, but yet provides a quantitative order,

corrected for the multiple testings performed for each gene.

The automatic system can also be used to guide the expert to investigate certain annotations or properties, when he or she uses it in parallel with his or her own prioritisation methodology. Moreover, it will be interesting for a geneticist to integrate his or her own data sources as well. The modular design of our software and the extensive usage of SOAP web services (14) could make this possible at two levels: (1) at the data level in case a web service can be built that returns either data vectors (microarray-like) or p values (GO-like) for a set of genes, and (2) at the rank level in case a text file with the geneticist's own ranking of the test genes can be provided, which can be integrated directly into the order statistics formula. For example, it will be interesting to incorporate phylogenetic information into the prioritisation. A straightforward way to achieve the inclusion of this particular data type, which will be made available in our software, is the inclusion of the "disease probabilities" of López-Bigas and Ouzounis (7). These are calculated from sequence and phylogeny information and they form a good summarised measure of a gene's phylogeny.

We have shown that all the used information sources contribute to the overall ranking, and that the correlation between individual models is minimal. The latter observation is useful for the selection of putatively relevant genes based on the p value of the order statistics, using a significance level. The equal weighting of all submodels at this stage did not cause any problems, but an automatic (or manual) weighting of individual submodels could be an interesting perspective for future work. This however would require a re-thinking of the order statistics technique, because the current formula does not allow to weigh the different sources differently. The observed bias towards unknown, or less characterised genes was kept to acceptable levels, and this bias will decrease further in the future, when new and better high-throughput data becomes available, and when the genome annotation and curation processes reach their finalisation.

In some previous studies on computational gene prioritisation, where only one or two data types are used, a case study was performed on recent findings of disease associated genes, for which the knowledge was not yet incorporated into the database (or alternatively as in (8), the text indexing was performed without the documents that literally described the association). In our case such a study is not feasible because of (1) the large number of data types and (2) the knowledge overlap between data types (e.g., many GO annotations are based on the literature). It is therefore intractable to determine which records out of which data sources have to be

removed to perform an honest study. We reckoned however that an original case study as described for cardiomyopathies, CHDs, and arrhythmias, with the use of all available information besides textual data, would nonetheless be a useful illustration of the capabilities of our method. In conclusion, we have presented for the first time a bioinformatics method for fast and automatic gene prioritisation that integrates many data types of different origins. The performance was unexpectedly encouraging to us, and so the strategy seems ready to be used for real gene hunting cases. Therefore we will make this easy to use software tool called ENDEAVOUR freely available, and we will use this software ourselves in further collaborative projects with human geneticists.

Data and methods

Text-mining: LocusLink and MEDLINE

TXTGate (15) is a text-mining application designed towards the analysis of the textual coherence of groups of genes. We use TXTGate's textual profiles of all genes in LocusLink, based on a Gene Ontology (GO) vocabulary (for a detailed description of the indexing we refer to (15)). A textual profile contains for each (stemmed) term of the GO vocabulary a weight describing the relative importance of this term for this gene. The textual data, which were used to calculate these weights, consist of all the MEDLINE abstracts linked to this gene via the PUBMED identifiers recorded in the LocusLink record. Gene prioritisation based on textual data is done by calculating the Pearson correlation between a test gene's textual profile and the average textual profile of the training genes. A high similarity between a test gene and the training genes means that the core of the literature abstracts that describe both, have a lot of terms in common, and they thus talk—in a general sense—about the same subject, no matter what the detailed messages in the abstract are. That is, textual profiles with contrasting statements that use the same words will still be similar. For example, if an abstract on gene x states that 'protein X stabilizes tau plaques' and an abstract on gene y states that 'protein Y solubilizes tau plaques', the textual profiles of gene x and gene y could be similar due to the common occurrence of the phrase 'tau plaques'. In the context of this approach (allowing prioritisation based on general similarity to an heterogeneous set of disease genes), this general measure of similarity may be an asset rather than a drawback.

Functional annotation: GO and KEGG

In those cases where the textual profiles could suffer from the noise in the literature data, the curated Gene Ontology (GO) data brings salvation. GO is a manually curated vocabulary that is used for the functional annotation of genes (16) and is structured as a hierarchical tree. Prioritisation is done by comparing the GO annotation of a test gene with the statistically over-represented GO terms in the training set (in fact, the set of actually annotated terms is extended and also incorporates all parents of the annotated terms up to the root of the tree). For example, if the proteins of most genes in a training set are in some way involved in linking cytoskeleton filaments to the plasma membrane, then GO terms like 'cytoskeleton' (GO:0005856),

'cytoskeletal anchoring' (GO:0007016), and so on, could be over-represented. If one of the test genes is annotated with any of these terms, it will get a high ranking according to the GO data. The KEGG database (17) is an even more structured source of functional annotation. It contains the members of known biological pathways. Similarly as for GO, we calculate whether certain pathways are over-represented in the training set and will give a good score (i.e., a low rank) to those test genes that are involved in one of the pathways that is important for the training set.

Protein information: InterPro and BIND

InterPro is a database of protein families, domains and functional sites (18). For each training gene the InterPro attributes are retrieved from the Ensembl Mart database (currently the ensembl_mart_22_1 database hosted at ensembldb.ensembl.org). An example of an InterPro attribute is IPR000418 (Name=Ets-domain) for which there are nineteen human proteins known to carry this domain. Scoring test genes using the InterPro protein domains is done by meta-analysis (see further). If a certain protein domain is over-represented in the training set as compared to the full genome, and if a test gene also carries this particular domain, then it will get a good ranking according to the InterPro data.

Another interesting data type that we use to score test genes is protein interaction data, for which we take data from the Biomolecular INteraction Database (BIND) (19). BIND contains interaction data from high-throughput experiments (for example, yeast two-hybrid assays) and from hand-curated information gathered from the scientific literature. The idea behind using protein interaction data for gene prioritisation is that one can expect a test gene to be more related to the training set if its protein directly interacts with one of the proteins of the training genes, or if it has a common interaction partner with one of them. In practice, all the proteins of the training genes and all their interaction partners are collected and the overlap between this set and the set containing a protein (encoded by a test gene) and its interaction partners is used to calculate the similarity score.

Gene expression: microarray data and ESTs

Microarray data has been used previously for gene prioritisation, for example in (20; 5; 6). We use the Atlas gene expression data of 47 normal human tissues (21). However, it is obvious that disease- or process-specific microarray data are more informative for particular training and test sets. For example, if a geneticist has performed his or her own microarray experiment that measures gene expression in healthy versus diseased patients (or if such data are available in public repositories, such as ArrayExpress or GEO), then a prioritisation based on these data is more likely to give good performance. Our modular implementation allows easy inclusion of any microarray data set into the prioritisation methodology (see also the Software availability section).
Next to microarray-based gene expression data, we reasoned that the large repositories of EST-based anatomical expression in the human body also contain valuable information that can be used for gene prioritisation. We use the EST-based expression data available via the Ensembl Mart database. As is done for GO, model training consists of calculating a p-value for each anatomical site that measures its statistical over-representation within the training set. Scoring a test gene with this EST-based model is done by meta-analyis (see below).

Cis-regulatory elements

In the prioritisation process we currently use *cis*-regulatory information in two different ways. Firstly, we compare all (offline) predicted instances of a library of transcription factor binding models (position weight matrices or PWMs), in all human-mouse conserved non-coding sequences (CNS) upstream of a test gene (10 kilobases), with the averaged instances of the training set. More information on this data set can be found in (3; 22). The predicted binding sites of all available transcription factors are recorded in a vector (for instance, of length 400 if there are 400 PWMs), where each element represents the best score of this PWM in all human-mouse conserved sequence blocks upstream of that gene. Comparison with the training vector is done by calculating the Pearson correlation.
Secondly, the best combination of three transcription factors within maximally 300 bp in the set of human-mouse CNSs is searched in the training set using the Genetic Algorithm version of our ModuleSearcher algorithm (22), using 20 generations. Scoring of a test gene is done by our ModuleScanner algorithm (3) that essentially sums up the best scores in all test gene CNSs of the

three PWMs of the trained model.

Sequence similarity: BLAST

There are examples of diseases that can be caused by proteins of the same family, for example Presenilin 1 and Presenilin 2 in Alzheimer's disease. The *e*-value of the BLAST between the (longest) coding sequences of these two genes is 10^{-133}, thus they are highly similar. One can imagine that a researcher would perform a BLAST search of a number of test genes and use his or her expert knowledge to judge whether the hits make sense. In the same sense we use a BLAST search to score test genes against a set of training genes. Judging whether a hit is relevant is done automatically by restricting the BLAST search on an ad hoc created BLAST database consisting of all coding sequences of the training set. Test genes that have a low significance value of the BLAST are similar to one of the training genes and will get a low rank.

Vector-based similarity measures

For information sources summarised using a vector representation, we use the Pearson correlation (in the case of microarray gene expression data and transcription factor binding sites)

$$\mathrm{r}_{\mathrm{Pearson}}(i,j) = \frac{\sum_q (x_{i,q} - \overline{x}_i)(x_{j,q} - \overline{x}_j)}{\sqrt{\sum_q (x_{i,q} - \overline{x}_i)^2}\sqrt{\sum_q (x_{j,q} - \overline{x}_j)^2}},$$

or the cosine similarity (in the case of textual information from MEDLINE)

$$\mathrm{sim}_{\cos}(i,j) = \frac{\sum_q x_{i,q} x_{j,q}}{\sqrt{\sum_q x_{i,q}^2}\sqrt{\sum_q x_{j,q}^2}}.$$

Meta-analysis

For the GO, KEGG, EST, and InterPro data types, the following meta-analysis is used to calculate a similarity score for a test gene compared to a set of training genes. For each gene in a set of training genes, all relevant attributes are collected (see before). Next, for each attribute a *p*-value is calculated using a binomial statistic that represents the statistical over-representation of this attribute within the training set.

Coherent training sets for any of the four characteristics will contain statistically significant p-values. When a group of test genes is scored using these data types, the p-values corresponding to the annotated attributes of a test gene are combined using Fisher's method ($\sum -2\log p_i$), generating a new p-value using the χ^2-distribution. The test genes are then ranked according to this new p-value.

Order statistics

Given the heterogeneity of the scoring results of individual information models (correlations, p-values, and counts), a meta(-meta)-analysis is not trivial since it would require a p-value for each submodel, and the calculation of p-values from correlation measures cannot be achieved straightforwardly. However, the scoring with each individual model results in N different rankings $r_1, r_2, ..., r_N$, one for each of the N data types used. Instead of directly combining the results of each submodel, we can combine the ranking according to the submodels using order statistics. The ranks are divided by the total number of ranked genes (excluding genes with no rank because of missing values) and a q-value is calculated that represents the probability of getting the observed rank ratios by chance. This q-value is calculated using the joint cumulative distribution of an N-dimensional order statistic as was also done by (23) (see `http://www.math.uah.edu/stat/urn/urn4.xml` for a description):

$$Q(r_1, r_2, ..., r_N) = N! \int_0^{r_1} \int_{s_1}^{r_2} \dots \int_{s_{N-1}}^{r_N} ds_N ds_{N-1} ... ds_1$$

Stuart *et al.* propose following recursive formula to efficiently compute above integral:

$$Q(r_1, r_2, ..., r_N) = \sum_{i=1}^{N} (r_{N-i+1} - r_{N-i}) Q(r_1, r_2, ..., r_{N-i}, r_{N-i+2}, ..., r_N),$$

where r_0=0. We noticed however that this formula is intractable for $N > 12$ because its complexity is O($N!$). We propose an alternative formula with complexity O(N^2):

$$V_k = \sum_{i=1}^{k-1} (-1)^{i-1} \frac{V_{k-i}}{i!} r_{N-k+1}^i$$

with V_0=1. The solution of the integral can be found by calculating V_N. Since the q-values calculated this way are not uniformly distributed, we have to fit a distribution for every possible number of ranks and use this distribution to estimate a p-value. We found that the q-values for $N <= 5$ randomly and uniformly drawn ranks are approximately distributed according to a Beta-distribution. For $N > 5$ the distributions can be modelled by a Gamma-distribution. The cumulative distribution function of these distributions provides us with a p-value for every q-value from the order statistics. Next to the original N rankings, we now have a $(N+1)^{th}$ which is the combined ranking resulting from our genomic data fusion.

Handling missing values

A major issue for gene prioritisation are missing values, because how can one judge the similarity between a test gene and a training set, if there is data missing? The order statistics provides us with a solution, because it is based on rank ratios rather than absolute ranks and because each probability can be computed with its particular value of N equal to the number of genes *with* information regarding a certain submodel (see Data and Methods Section). For genes without information regarding a certain submodel, no comparison can be made between a test gene and the submodel and thus no rank can be given. Hence they are not taken into account when applying the order statistics.

Genes from `TEST` for which there *is* data available, but for which there is no similarity with the training set also have to be ranked with caution. Such genes have the highest (i.e., worst) possible score for a particular submodel (for instance, 1.0 for the GO submodel) are very dissimilar to `TRAIN` according to this submodel. Imagine a case where all test genes have the same extreme score of 1.0. Since they all have the same score they will all get the same (best) rank of 1 and the order statistics will put them high in the overall ranking. To avoid this problem all genes with maximal dissimilarity get the maximum rank (which is equal to the number of genes in `TEST`).

Rank ROC curves

The results of the cross-validation can be visualised in a Rank ROC (Receiver Operating Characteristic) curve, where the y-axis represents the sensitivity (i.e., the proportion of true

positives) and the x-axis represents one minus the specificity (i.e., the proportion of true negatives):

$$\text{sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}}$$

We have called this Rank ROC because we do not perform a traditional classification with one model, but multiple prioritisations with different models. The values in the above formulas are calculated from all iterations of all diseases and their interpretation is the following: (1) the number of true positives (TP) is the number of times that the left-out gene is ranked above the cut-off; (2) the false positives (FP) are all genes besides the left-out gene that are ranked above the cut-off (these can be thought of as being retained for further evaluation but they are probably not disease associated); (3) the true negatives (TN) are those genes that are ranked below the cut-off and that are not the left-out gene; and (4) the number of false negatives (FN) is the number of times that the left-out gene is ranked below the cut-off (in these cases the real disease associated gene is not retained for potential further analysis). In a Rank ROC curve as in Figure 3, the sensitivity and (1-specificity) are plotted for each possible cut-off value. On the one hand, such curves can be used to choose a cut-off value (giving a desirable balance between FP and FN) and on the other hand they can be used to compare different kinds of prioritisations. The area under the curve (AUC) is a measure of the performance, which in our case would equal one if every left-out gene (i.e., the wanted disease gene in each test set) is ranked first for all tested diseases. The AUC would be 0.5 if the prioritisation is not better than ranking the genes randomly. Although the Rank ROC is not an ROC per se, it is an appropriate measure of the proportion of genes correctly (incorrectly) included (or left-out) from a list of follow-up genes—as a function of the length of such a follow-up list.

Acknowledgements

We wish to thank all groups and consortia that made their data freely available: Ensembl, NCBI (LocusLink and MEDLINE), Gene Ontology, BIND, KEGG, Atlas, InterPro, BioBase (public release of TRANSFAC at www.gene-regulation.com), and the Disease Probabilities from López-Bigas and Ouzounis (7). We also thank Patrick Glenisson for help with the text-mining component, Joke Allemeersch and Gert Thijs for their advice on the order statistics and Peter Van Loo for testing our software within a wet-lab setting. This work is partially supported by Instituut voor de aanmoediging van Innovatie door Wetenschap en Technologie in Vlaanderen (IWT) projects STWW-00162, STWW-Genprom, GBOU-SQUAD-20160; Research Council KULeuven GOA Mefisto-666, GOA-Ambiorics, IDO genetic networks; and FWO: Fund for Scientific Research-Flanders (Belgium) projects G.0115.01 and G.0413.03; IUAP V-22 (2002-2006).

Figure legends

Figure 1

Candidate gene prioritisation based on multiple data types according to the similarity with a set of training genes. Currently the available data types (described in the Data and methods section) are: Textual data (Te), Microarray gene expression (Ma), Gene Ontology annotations(Go), BIND protein interactions (Bi), transcription factor binding sites or Motifs (Mo), *Cis*-regulatory modules (Cr), EST-based expression (Es), KEGG pathways (Ke), BLAST-based sequence similarity (Bl), and InterPro protein domains (Ip). There are two Ma models in the figure, illustrating the possibility of using multiple microarray gene expression data sets. The overall ranking of gene_i shown as a white square is calculated from all individual rankings according to the different data types.

Figure 2

The cross-validation procedure used to validate our computational prioritisation of candidate disease genes based on their 'similarity' with a set of training genes.

Figure 3

Rank ROC curves for the cross validation. The genes for 29 diseases selected from OMIM are used as training sets. Each gene in each disease is used once as a left-out gene and scored together with 9, 49, or 99 randomly chosen genes on the model trained on the remaining left-in genes.

Figure 4

Rank ROC curves for 50 test genes among which is 1 'left-out gene'. Each Rank ROC curve represents one submodel. (A) 29 OMIM disease gene sets were used as training sets according to Figure 2. (B) 5 sets of 20 random genes were used as training sets.

Figure 5

Spearman correlation between the different submodels. Between the rankings of different data types and between the rankings of the individual data types and the overall ranking. Ra overall rank; Go Gene Ontology; Es EST-based expression; Ip Interpro domains; Ke KEGG pathways; Bi BIND protein interactions; Te text-mining; Mo motifs or transcription factor binding sites; Cr *cis*-regulatory modules (combinations of motifs); At Atlas microarray expression data; Bl sequence similarity using BLAST

Figure 6

Bias to well-characterised genes. 29 sets of 50 random test genes were prioritised based on 29 OMIM disease training sets. (A) The number of available information models is plotted against the precentage of genes within each ranking interval. (B) The bars represent the percentage of genes within each ranking interval for genes with a HUGO symbol (1) and genes without HUGO symbol (2).

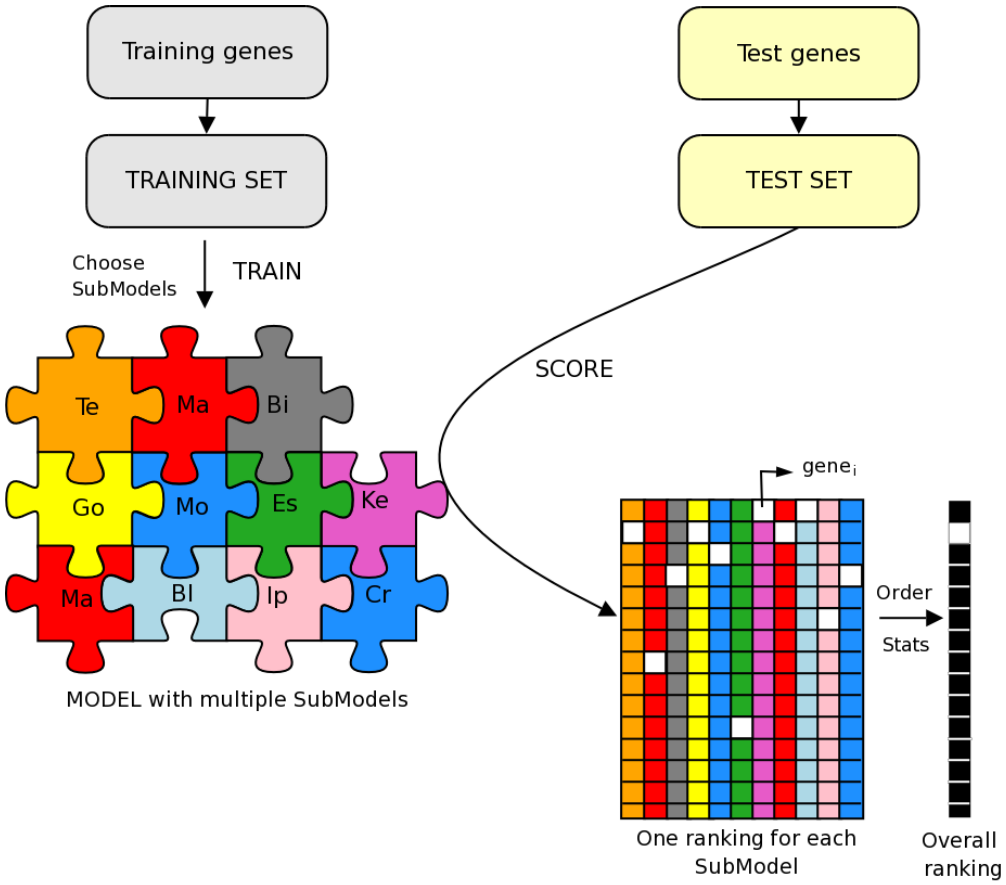

Training genes
TRAINING SET
Choose SubModels
TRAIN
Test genes
TEST SET
SCORE
Te
Ma
Bi
Go
Mo
Es
Ke
Ma
Bl
Ip
Cr
MODEL with multiple SubModels
gene$_i$
Order
Stats
One ranking for each SubModel
Overall ranking

Figure 1:

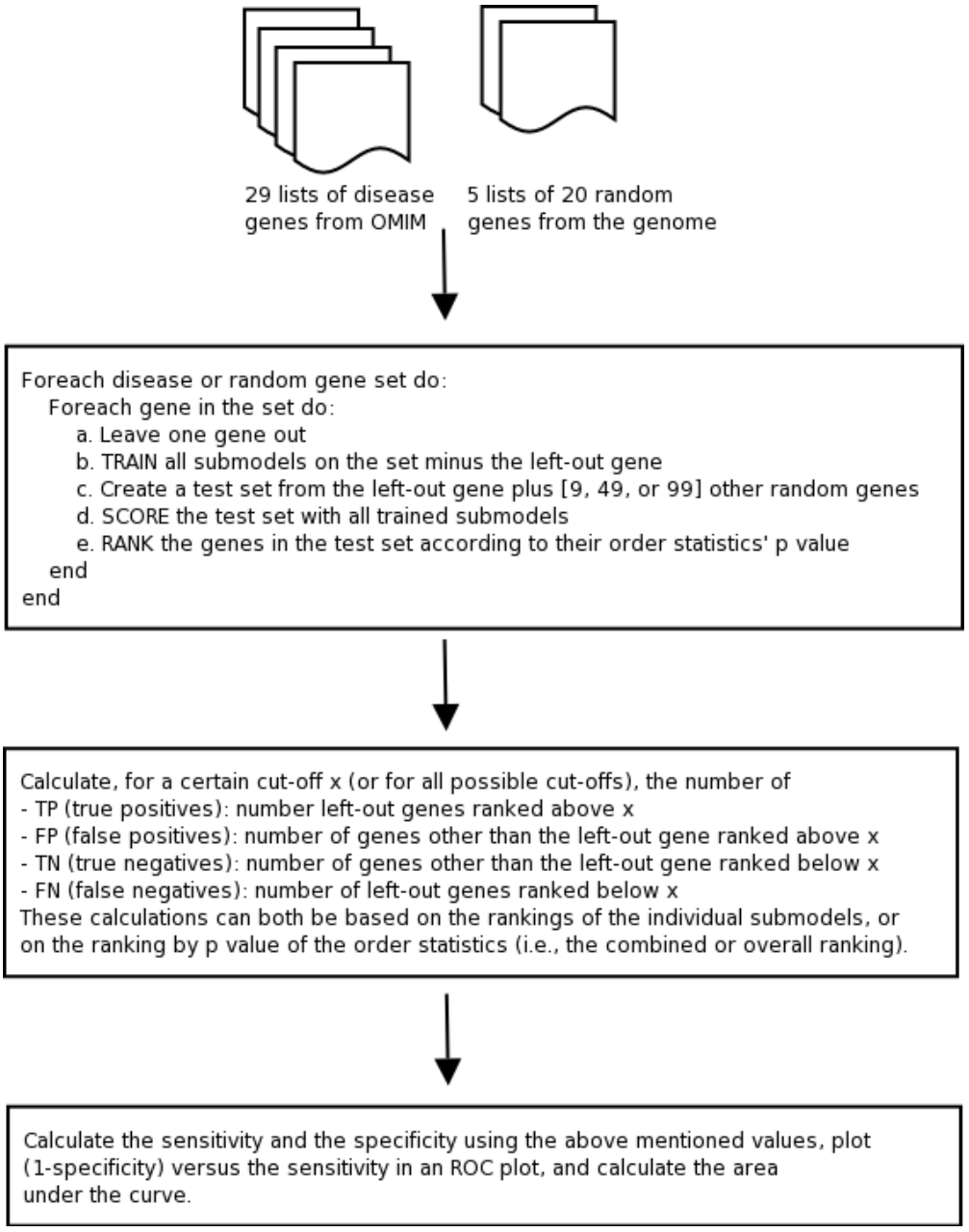

29 lists of disease genes from OMIM
5 lists of 20 random genes from the genome
Foreach disease or random gene set do:
Foreach gene in the set do:
a. Leave one gene out
b. TRAIN all submodels on the set minus the left-out gene
c. Create a test set from the left-out gene plus [9, 49, or 99] other random genes
d. SCORE the test set with all trained submodels
e. RANK the genes in the test set according to their order statistics' p value
end
end
Calculate, for a certain cut-off x (or for all possible cut-offs), the number of
- TP (true positives): number left-out genes ranked above x
- FP (false positives): number of genes other than the left-out gene ranked above x
- TN (true negatives): number of genes other than the left-out gene ranked below x
- FN (false negatives): number of left-out genes ranked below x
These calculations can both be based on the rankings of the individual submodels, or on the ranking by p value of the order statistics (i.e., the combined or overall ranking).
Calculate the sensitivity and the specificity using the above mentioned values, plot (1-specificity) versus the sensitivity in an ROC plot, and calculate the area under the curve.

Figure 2:

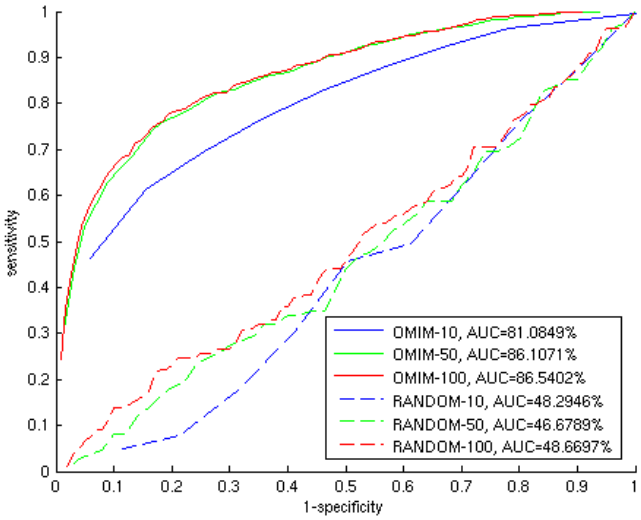

sensitivity
1-specificity
OMIM-10, AUC=81.0849%
OMIM-50, AUC=86.1071%
OMIM-100, AUC=86.5402%
RANDOM-10, AUC=48.2946%
RANDOM-50, AUC=46.6789%
RANDOM-100, AUC=48.6697%

Figure 3:

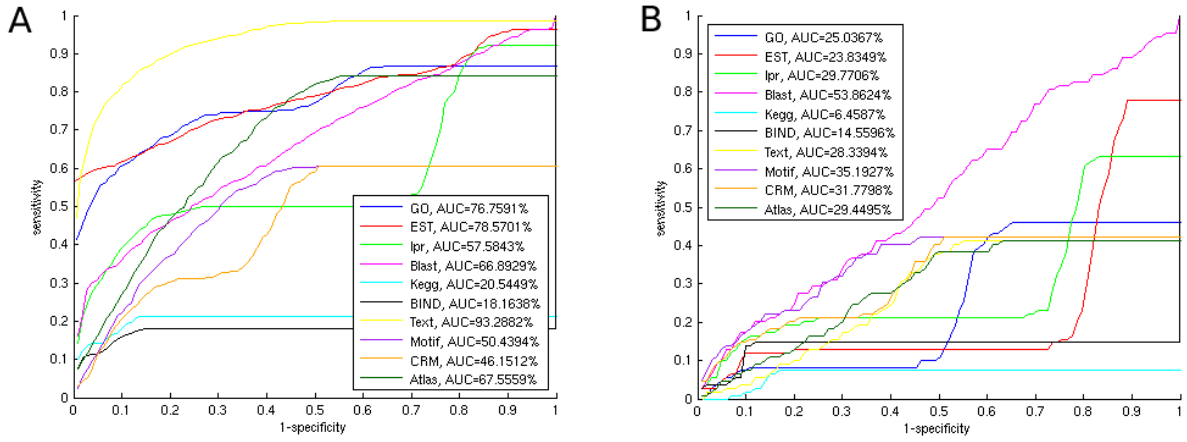

A
sensitivity
1-specificity
GO, AUC=76.7591%
EST, AUC=78.5701%
lpr, AUC=57.5843%
Blast, AUC=66.8929%
Kegg, AUC=20.5449%
BIND, AUC=18.1638%
Text, AUC=93.2882%
Motif, AUC=50.4394%
CRM, AUC=46.1512%
Atlas, AUC=67.5559%
B
sensitivity
1-specificity
GO, AUC=25.0367%
EST, AUC=23.8349%
lpr, AUC=29.7706%
Blast, AUC=53.8624%
Kegg, AUC=6.4587%
BIND, AUC=14.5596%
Text, AUC=28.3394%
Motif, AUC=35.1927%
CRM, AUC=31.7798%
Atlas, AUC=29.4495%

Figure 4:

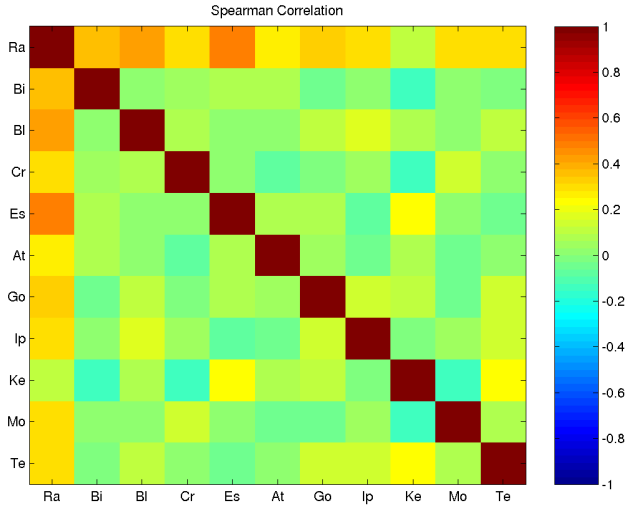

Spearman Correlation
Ra
Bi
Bl
Cr
Es
At
Go
Ip
Ke
Mo
Te
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1

Figure 5: .

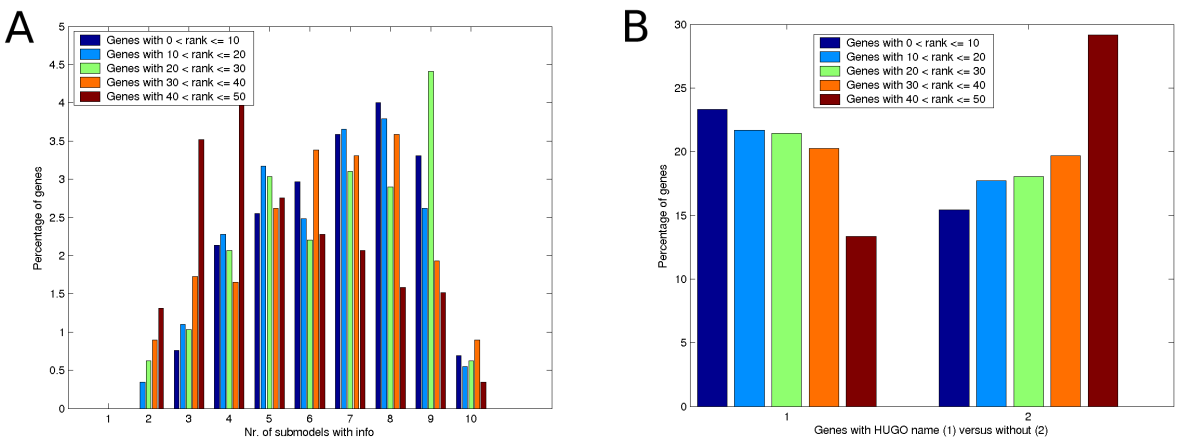

A
Genes with 0 < rank <= 10
Genes with 10 < rank <= 20
Genes with 20 < rank <= 30
Genes with 30 < rank <= 40
Genes with 40 < rank <= 50
Percentage of genes
Nr. of submodels with info
B
Genes with 0 < rank <= 10
Genes with 10 < rank <= 20
Genes with 20 < rank <= 30
Genes with 30 < rank <= 40
Genes with 40 < rank <= 50
Percentage of genes
Genes with HUGO name (1) versus without (2)

Figure 6:

References

1. Johansson, O., Alkema, W., Wasserman, W., and Lagergren, J. 2003. Identification of functional clusters of transcription factor binding motifs in genome sequences: the MSCAN algorithm. *Bioinformatics,* **19**(Suppl 1), I169–I176.

2. Sharan, R., Ovcharenko, I., Ben-Hur, A., and Karp, R. 2003. CREME: a framework for identifying *cis*-regulatory modules in human-mouse conserved segments. *Bioinformatics,* **19**(Suppl 1), I283–I291.

3. Aerts, S., Van Loo., P., Thijs, G., Moreau, Y., and De Moor., B. 2003. Computational detection of cis-regulatory modules. *Bioinformatics,* **19 Suppl 2**, II5–II14.

4. Botstein, D. and Risch, N. 2003. Discovering genotypes underlying human phenotypes: past successes for mendelian disease, future approaches for complex disease. *Nat Genet,* **33 Suppl**, 228–37.

5. Hauser, M., Li, Y., Takeuchi, S., Walters, R., Noureddine, M., Maready, M., Darden, T., Hulette, C., Martin, E., Hauser, E., Xu, H., Schmechel, D., Stenger, J., Dietrich, F., and Vance, J. 2003. Genomic convergence: identifying candidate genes for Parkinson's disease by combining serial analysis of gene expression and genetic linkage. *Hum Mol Genet,* **12**(6), 671–7.

6. Franke, L., Bakel, H., Diosdado, B., Belzen, M., Wapenaar, M., and Wijmenga, C. 2004. TEAM: a tool for the integration of expression, and linkage and association maps. *Eur J Hum Genet,* **12**(8), 633–8.

7. López-Bigas, N. and Ouzounis, C. 2004. Genome-wide identification of genes likely to be involved in human genetic disease. *Nucleic Acids Res,* **32**(10), 3108–14.

8. Perez-Iratxeta, C., Bork, P., and Andrade, M. 2002. Association of genes to genetically inherited diseases using data mining. *Nat Genet,* **31**(3), 316–9.

9. Freudenberg, J. and Propping, P. 2002. A similarity-based method for genome-wide prediction of disease-relevant human genes. *Bioinformatics,* **18 Suppl 2**, S110–5.

10. Robinson, S., Morris, C., Goldmuntz, E., Reller, M., Jones, M., Steiner, R., and Maslen, C. 2003. Missense mutations in CRELD1 are associated with cardiac atrioventricular septal defects. *Am J Hum Genet,* **72**(4), 1047–52.

11. Olson, T. and Keating, M. 1996. Mapping a cardiomyopathy locus to chromosome 3p22-p25. *J Clin Invest,* **97**(2), 528–32.

12. Ahmad, F., Li, D., Karibe, A., Gonzalez, O., Tapscott, T., Hill, R., Weilbaecher, D., Blackie, P., Furey, M., Gardner, M., Bachinski, L., and Roberts, R. 1998. Localization of a gene responsible for arrhythmogenic right ventricular dysplasia to chromosome 3p23. *Circulation,* **98**(25), 2791–5.

13. Weiss, R., Barmada, M., Nguyen, T., Seibel, J., Cavlovich, D., Kornblit, C., Angelilli, A., Villanueva, F., McNamara, D., and London, B. 2002. Clinical and molecular heterogeneity in the Brugada syndrome: a novel gene locus on chromosome 3. *Circulation,* **105**(6), 707–13.

14. Stein, L. 2002. Creating a bioinformatics nation. *Nature,* **417**(6885), 119–120.

15. Glenisson, P., Coessens, B., Van Vooren, S., Mathys, J., Moreau, Y., and De Moor, B. 2004. TXTGate: profiling gene groups with text-based information. *Genome Biol,* **5**(6), R43.

16. Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Dolinski, K., Dwight, S. S., Eppig, J. T., Harris, M. A., Hill, D. P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J. C., Richardson, J. E., Ringwald, M., Rubin, G. M., and Sherlock, G. 2000. Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat Genet,* **25**(1), 25–29.

17. Kanehisa, M., Goto, S., Kawashima, S., Okuno, Y., and Hattori, M. (2004) The KEGG resource for deciphering the genome.. *Nucleic Acids Res,* **32**(1), D277–80.

18. Mulder, N., Apweiler, R., Attwood, T., Bairoch, A., Barrell, D., Bateman, A., Binns, D., Biswas, M., Bradley, P., Bork, P., Bucher, P., Copley, R., Courcelle, E., Das, U., Durbin, R., Falquet, L., Fleischmann, W., Griffiths-Jones, S., Haft, D., Harte, N., Hulo, N., Kahn, D., Kanapin, A., Krestyaninova, M., Lopez, R., Letunic, I., Lonsdale, D., Silventoinen, V., Orchard, S., Pagni, M., Peyruc, D., Ponting, C., Selengut, J., Servant, F., Sigrist, C., Vaughan, R., and Zdobnov, E. 2003. The InterPro Database, 2003 brings increased coverage and new features. *Nucleic Acids Res,* **31**(1), 315–8.

19. Bader, G., Betel, D., and Hogue, C. 2003. BIND: the Biomolecular Interaction Network Database. *Nucleic Acids Res,* **31**(1), 248–50.

20. vanDriel, M., Cuelenaere, K., Kemmeren, P., Leunissen, J., and Brunner, H. 2003. A new web-based data mining tool for the identification of candidate genes for human genetic disorders. *Eur J Hum Genet,* **11**(1), 57–63.

21. Su, A., Cooke, M., Ching, K., Hakak, Y., Walker, J., Wiltshire, T., Orth, A., Vega, R., Sapinoso, L., Moqrich, A., Patapoutian, A., Hampton, G., Schultz, P., and Hogenesch, J. 2002. Large-scale analysis of the human and mouse transcriptomes. *Proc Natl Acad Sci U S A,* **99**(7), 4465–4470.

22. Aerts, S., Van Loo, P., Moreau, Y., and De Moor, B. 2004. A genetic algorithm for the detection of new cis-regulatory modules in sets of coregulated genes. *Bioinformatics,* **20**(12), 1974–6.

23. Stuart, J., Segal, E., Koller, D., and Kim, S. 2003. A gene-coexpression network for global discovery of conserved genetic modules. *Science,* **302**(5643), 249–55.