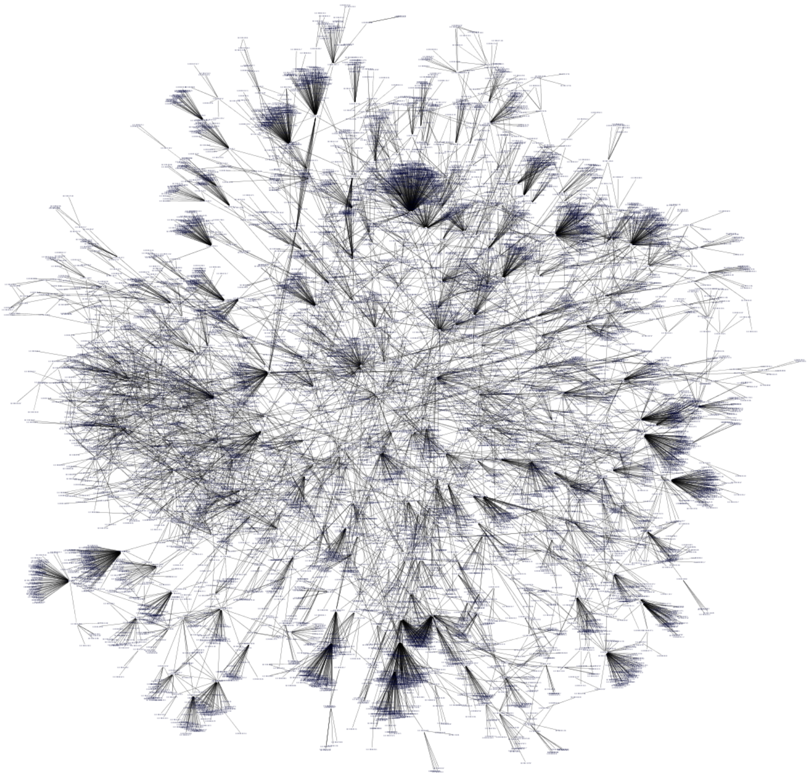
Universitat de Barcelona

Methods and Models for the Analysis of Biological Significance Based on High-Throughput Data

Jose Luis Mosquera Mayo

Methods and Models for the Analysis of Biological Significance Based on High Throughput Data



Jose Luis Mosquera Mayo

UNIVERSITAT DE BARCELONA

Methods and Models for the Analysis of Biological Significance Based on High Throughput Data

Mètodes i Models per a l'Anàlisi de la Significació Biològica Basada en Dades d'Alt Rendiment

Memòria presentada per en Jose Luis Mosquera Mayo
per optar al grau de doctor per la Universitat de Barcelona

SIGNATURES

El Doctorand:
Jose Luis Mosquera Mayo

El Director de la Tesi:
Dr. Alexandre Sánchez Pla

El Tutor de la Tesi:
Dr. Josep María Oller Sala

Programa de doctorat en estadística
Departament d'Estadística, Facultat de Biologia
Universitat de Barcelona

Acknowledgements

"A teacher affects eternity, he can never tell where his influence stops."

Henry Brooks Adams

"Mathematicians do not study objects, but relations between objects."

Jules Henri Poincaré

It is truly an honor for me to address these words of gratitude to everyone who has contributed, to a greater or lesser extent by supporting me during this exciting program of studies, to help make my lifelong dream come true: to complete my doctoral dissertation.

First and foremost, I would like (and feel compelled) to mention a special debt of gratitude to **Prof. Alex Sánchez Pla**. I am enormously grateful to him, as the director of this dissertation, for having taught, guided and supported me, correcting my scientific work with an interest and dedication that have greatly exceeded all the expectations that I, as a UB student, could have had. Likewise, I would also like to thank him once again for his work as the head of the Statistics and Bioinformatics Unit at the VHIR, for directing, supervising and correcting all the projects in which I have participated over the last six years as a senior technician in the department.

It is also my deep personal desire to thank all the members of the Statistical Research and Bioinformatics Group, as well as the professors in the UB Statistics Department, for everything they have taught me, both professionally and as a human being. I learned something from them each and every day. For these reasons, I would like to specifically express my thanks to **Prof. M. Carme Ruíz de Villa**, **Prof. Francesc Carmona**, **Prof. Esteban Vegas**, **Prof. Ferran Reverte**r and **Prof. Antoni Mińarro**. I would also like to recognize my office mate and sparring partner, both at the UB and the VHIR, **Mr. Israel Ortega**. I would not wish to neglect a big thank you to the department secretaries, **Ms. Roser Maldonado**

and **Ms. Elisabet Ballester**, for their hard work and dedication; without them, the bureaucracy would have been a disaster.

I would like to extend these thanks to all the members of the Statistics and Bioinformatics Unit and the High Technology Unit at the VHIR. In particular, I would like to thank **Mr. Alejandro Artacho**, **Dr. Ricardo Gonzalo** and **Dr. Francisca Gallego** for having shared their knowledge and friendship with me. I must not forget a few words of appreciation for the researchers at the VHIR, with whom I have collaborated closely, and who without knowing it, have at times indirectly contributed to this dissertation. I would therefore like to especially mention **Dr. María Vicario**, **Dr. Cristina Martínez**, **Dr. Javier Santos**, **Dr. Rosanna Paciucci**, **Dr. Andreas Doll** and **Dr. Jaume Reventós**.

Finally, I would like to express my eternal gratitude to **Antonio**, **Carmen**, **Chus**, **María**, **Magdalena**, my parents, my brother and my grandparents, who are no longer with us; without them, I would never have been able to complete this project. They have always been with me, through thick and thin. Furthermore, I would like to sincerely thank **Inma**, **Ricardo**, **Jordi**, **Ferran**, **Mireia** and **Paula** for having put up with me, without asking for anything in return.

Thank you very much to all those whom I have not mentioned, who have contributed in some way to this personal achievement; please forgive me for this omission. Thank you.

Contents

List of Tables

List of Figures

Glossary

BP Biological Process. 27, 28, 31–33, 59, 66, 109, 124

CC Cellular Component. 27, 31–33, 59, 66, 109, 124

CGH Comparative Genomic Hybridization. 130

DAG Directed Acyclic Graph. 32, 33, 51, 55–57, 59, 62, 63, 65–67, 69, 72, 78–80, 84–86, 88, 94, 96, 102–104, 109

DCA Disjoint Common Ancestors. 37, 76

ES Enrichment Score. 41

GO Gene Ontology. 25–27, 29–33, 38–40, 43–45, 49–52, 59, 61, 62, 64–67, 72, 75, 77–80, 84, 86–88, 95, 96, 98, 99, 101, 102, 104, 105, 109–119, 122–125, 129–133, 135–137, 144–146, 151, 152, 154–157, 162, 164, 172, 174, 175, 178–181, 185–187, 191–193, 196–198, 201–206

GSEA Gene Set Enrichment Analysis. 38, 40

HTML HyperText Markup Language. 126, 127

IC Information Content. 37, 76–80, 84, 86, 88, 95–97, 102, 103

KB Knowledge Base. 125

KEGG Kyoto Encyclopedia of Genes and Genomes. 26, 39, 40

LCA Lower Common Ancestors. 96, 97

LODE Live OWL Documentation Environment. 190

Loess Locally Weighted Regression. 117, 118, 162, 203

MDS Multidimensional Scaling. 118, 120, 121, 174, 175, 180–186, 203–205

MEA Modular Enrichment Analysis. 38–40

MF Molecular Function. 27, 29, 31–33, 59, 66, 109, 124

MICA Most Informative Common Ancestor. 37, 76, 78, 79, 88, 95–97, 102

MSigDB Molecular Signatures Database. 40

OOC Object-Ontology Complex. 65–67, 72, 78, 79, 82, 87, 89, 96, 102–104

OWL Web Ontology Language. 127, 188–192

PAM Partitioning Around Medoids. 119, 172, 175

POSET Partially Ordered Set. 67, 69, 72, 80, 81, 87, 92, 101, 103

POSO Partially Ordered Set Ontology. 72, 80, 87, 103

PRO PROtein Ontology. 49

RDF Deoxyribonucleic acid. 126, 127

SAGE Serial Analysis of Gene Expression. 130

SC Silhouette Coefficient. 172, 174

SEA Singular Enrichment Analysis. 38–40

SO Sequence Ontology. 49

UPGMA Unweighted Pair Group Method with Arithmetic Mean. 119

URI Uniform Resource Identifier. 125, 127

XML eXtensible Markup Language. 126, 127

Part I

Introduction and Objectives

Chapter 1

Introduction: Scope of the Thesis

Throughout most of the twentieth century, molecular biology has been reductionist. Reductionism is an approach to understanding complex systems (e.g. cells, tissues, or organisms) by reducing them to simpler or more fundamental molecules (e.g. individual genes, mRNAs, proteins or metabolites). However, during the last twenty years, there has been a change of strategy on the subject that is totally different. It consists in study all the molecules in one or a population of complex systems (e.g. genome, transcriptome, proteome or metabolome), that is, studying a complex system by taking a holistic point of view of the molecules that make up such a complex system. This revolution began with the deciphering of the whole genome sequences of several organisms —among them the human genome—, and rapidly, similar ideas were applied to the study of the transcriptome, proteome and metabolome. This resulted in the emergence of *omic* studies: genomics, transcriptomics, proteomics and metabolomics (figure 1.1).

With the advent of the omic era huge quantities of information have been generated, and it has become a major breakthrough for molecular biology. This has been possible thanks to a new generation of high-technologies known as *high-throughput* technologies. These technologies allow the performance, in a routine way, of new types of experiments to analyze simultaneously the behavior of thousands of features (e.g. genes, mRNA, proteins or metabolites) under different conditions.

There are different types of high-throughput technologies (e.g. microarrays, next generation sequencing and mass spectrometry) that allow the performance of a broad range of *omic experiments*. For example, a microarray experiment makes it possible to determine the correlation between the expression of a large list of genes, or proteins or entire genotypes of the phenotypic traits, characterizing a studied group. In the case of a Next

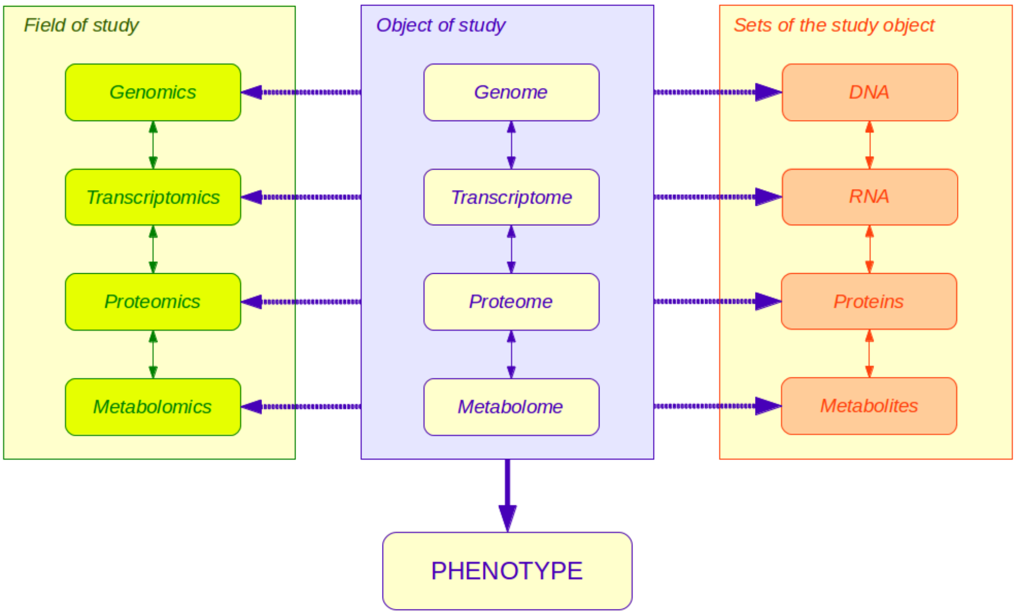


Figure 1.1: General schema showing the relationships between the genome, transcriptome, proteome and metabolome in omics studies.

Generation Sequencing (NGS) experiment, it is possible to obtain the genomic sequence of a new organism by assembling small “pieces”’ of DNA, or identifying genes that are being expressed in a particular organism, or analyzing protein interactions with DNA by identifying the binding sites of DNA associated proteins. And finally, in a Mass Spectrometry (MS) experiment a researcher can separate proteins, peptides or metabolites, according to their molecular mass and/or structure and so detect them.

Independently of the high-throughput technology used in an omic experiment, very often it will result in long lists of features which have been selected using some criteria to assign them statistical significance. For instance, in a microarray experiment a t-test can be used to identify genes differentially expressed between two or more conditions. With those lists in hand, a researcher is faced with the problem of finding a biological interpretation. However, most of the time, biological interpretation of a list of genes is not obvious. Even a biologist with great experience may have some difficulties in interpreting what the list of features mean. At this point, the experimenter has different alternatives. For instance, he/she could do a comprehensive search in the literature regarding each feature and collect information about whether a specific feature has a known function, or if some of them have been described to work in a cluster, or if this list of features is known to be associated with a phenotype under certain experimental conditions, and so on. But even if it is still possible to perform this task, it is almost certain that this quest will take a long period of time. Moreover, sometimes the number of items selected as being statistically significant is very high and it seems reasonable to (try to) summarize them by looking at what the list means from a biological point of view. Sometimes, instead, the selected items do not show any statistical significance, but even so it is expected -or it seems clear- that, biologically, they "mean something", probably related to the process being analyzed.

In whatever of the previous situations we find ourselves, the usual way to proceed is to shift the focus from "statistical" to "biological" significance. However, while there is a clear agreement about what statistical significance means, there is no consensus definition for biological significance at all.

1.1 Meaning of Biological Significance Concept

Interestingly, what many authors do to define *Biological Significance* is to redefine it in terms of statistical significance. This can be clearly seen in [41], where the authors state:

> ... *to understand the biological relevance of statistical differences in gene expression data* by examining significant differences in the distribution of Gene Ontology (GO) terms related to biological processes or molecular function.

This is not however the only possible definition. For instance, `GeneSifter` [61], a company presenting their goals as to "make it easier to understand the biological significance of your microarray data" does not give any definition of the term. The nearest explanation of what they mean is:

> ... *to characterize the biology involved in a particular experiment*, and to identify particular genes of interest... combining the identification of broad biological themes with the ability to focus on a particular gene...

In any case, it is clear that whatever they mean by Biological Significance they do not relate this concept to Statistical Significance.

Therefore, while many efforts are addressed to attribute Biological Significance to the results yielded in omic experiments, because it is of course an important step towards answering the biological questions that are being pursued by experiments, fewer efforts are devoted to clarifying what this concept exactly means.

1.2 The Gene Ontology

Attempts to perform a biological interpretation of results from high-throughput experiments are often based on existing annotation spaces (e.g. Ensembl [56], KEGG [86] or UniProt [35]). Many of these annotation spaces are focused on species or context-dependent. That is, the biological knowledge is associated with a specific species or biological vocabulary. Moreover, many of these resources use methods to manage the information that might only be properly used by specialists in a restricted domain, or sometimes these methods are very difficult to understand, or they are not appropriated, etc. Whatever the case, users are often not able to deal and share with the information stored in annotation spaces, because they are not able to access the biological knowledge. Furthermore, there are resources that store the information as plain text, and here it is very difficult to automate management because there are no recognition patterns to identify fields or traits associated with the information stored. That is, the generalization or extension to related situations is not straightforward. For these reasons and in order to facilitate the comparison of annotations and also to increase the unification of the biological knowledge, the scientific community developed a resource which does not depend on either specific organisms or experiments. This tool is the *Gene Ontology* (or commonly called *GO*) ([148]). It has become one of the most successful resources used for performing biological interpretations.

The GO is an annotation resource created and maintained by a public consortium, The Gene Ontology Consortium [1] ([152]), whose main goal is, *citing their mission, to produce a controlled vocabulary that can be applied to all organisms even as knowledge of gene and protein roles in cells is accumulating and changing.*

[1] The GO Consortium is a collection of biological databases and research groups actively involved in the GO project. This includes a number of model organism databases and multi-species protein databases, software development groups, and a dedicated editorial office.

In order to give a more comprehensive explanation of the GO project, the following subsections provide a comprehensive explanation of the information content, the scope, and the structure of the GO. Descriptions have been extracted and adapted from the GO documentation [151].

1.2.1 The Ontology Domains of the GO

The GO project provides an ontology of terms representing gene product properties. It is organized covering three domains:

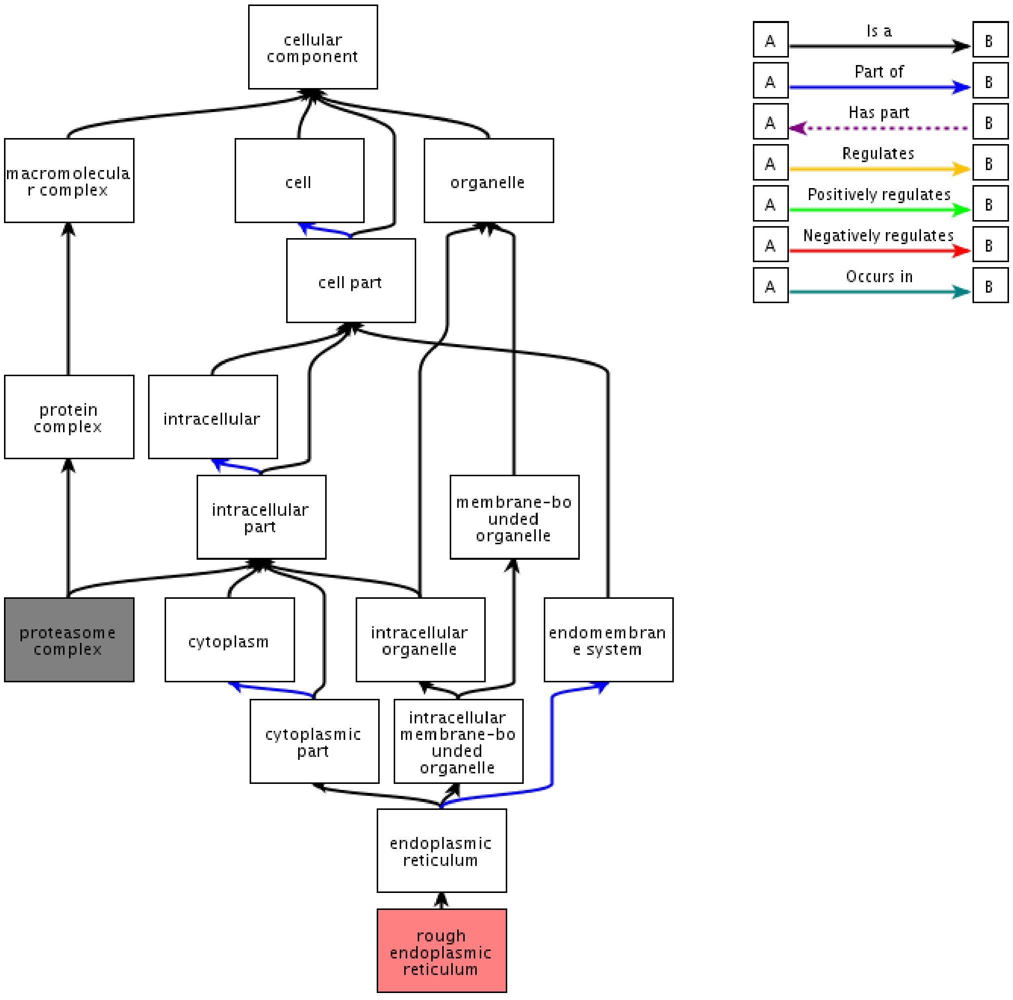
- *Cellular Component* (*CC*), the parts of a cell or its extracellular environment;
- *Biological Process* (*BP*), operations or sets of molecular events with a defined beginning and end, pertinent to the functioning of integrated living units: cells, tissues, organs, and organisms; and
- *Molecular Function* (*MF*), the elemental activities of a gene product at the molecular level, such as binding or catalysis.

One way to understand this organization is to think that individuals (i.e. gene products) have different tasks (i.e. functions) and they work together to achieve different goals (i.e. processes).

Currently, there are approximately 40.000 terms in the GO. More specifically: the Biological Process (BP) domain consists of 27.224 GO terms, Molecular Function (MF) domain consists of 10.725 GO terms, and Cellular Component (CC) domain consists of 3.745 GO terms. These numbers were extracted from the tool `AmiGO2` ([22]) when writing this thesis.

1.2.1.1 Cellular Component

The CC ontology describes the components of a cell, at the levels of subcellular structures and macromolecular complexes. This may be an anatomical structure (e.g. rough endoplasmic reticulum or nucleus) or a gene product group (e.g. proteasome complex, ribosome, or a protein dimer). A gene product *is located in* or *is a subcomponent of* a particular cellular component. Terms in CC ontology include multi-subunit enzymes and other protein complexes, but not individual proteins or nucleic acids, or multicellular anatomical terms (figure 1.2).

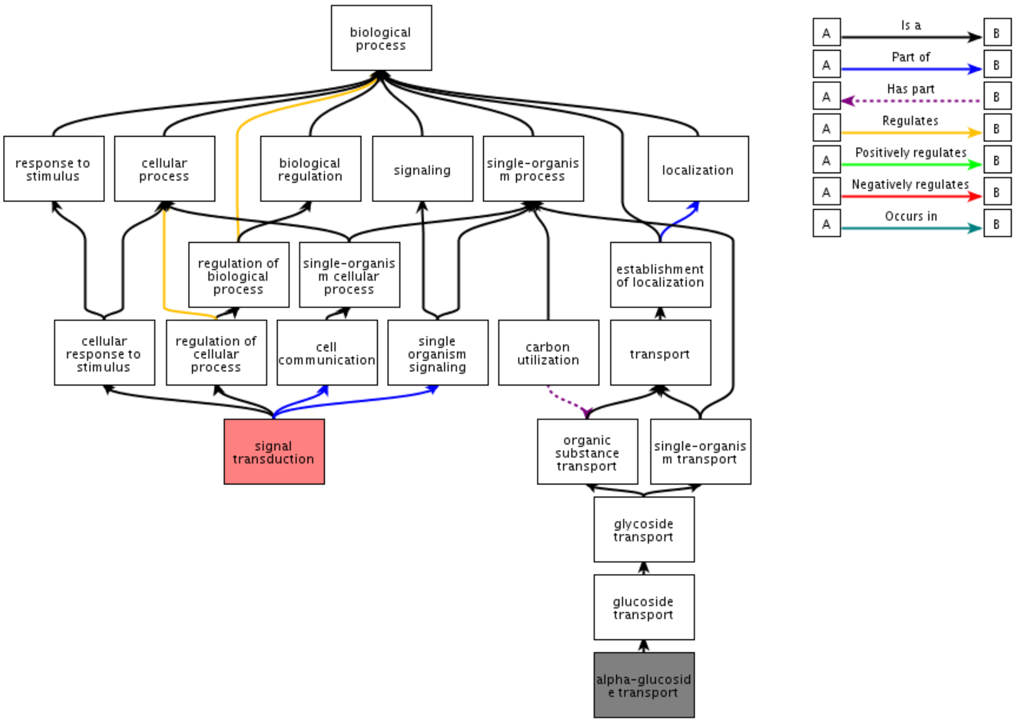


QuickGO - http://www.ebi.ac.uk/QuickGO

Figure 1.2: GO annotation terms of the Cellular Component domain for Proteasome complex (gray) and Rough Endoplasmic Reticulum (pink) terms. Colors of the links establish the types of relationships between two categories.

1.2.1.2 Biological Process

The BP ontology describes terms that represent collections of molecular events with a defined beginning and end. For instance, a broad biological process term is a cellular physiological process or a signal transduction, and an example of a more specific term is alpha-glucoside transport or pyrimidine metabolic process (figure 1.3).



QuickGO - http://www.ebi.ac.uk/QuickGO

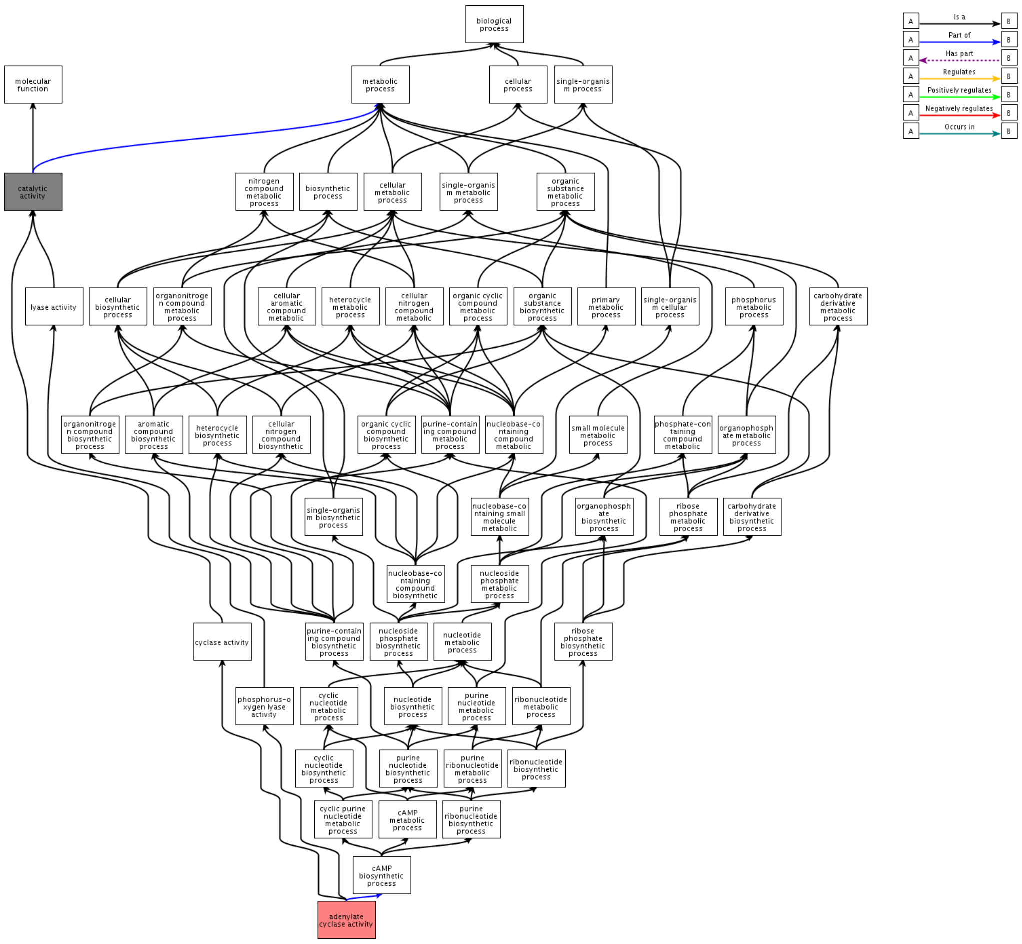
Figure 1.3: GO annotation terms of the Biological Process domain for Signal Transduction (pink) and Alpha-Glucoside Transport (gray) terms. Colors of the links establish the types of relationships between two categories.

It can be difficult to distinguish between a biological process and a molecular function, but in general a process must have more than one distinct steps.

A biological processes is not equivalent to a pathway. GO does not try to represent the dynamics or dependencies that would be required to fully describe a pathway.

1.2.1.3 Molecular Function

The MF ontology terms represent activities rather than the entities (molecules or complexes) that perform the actions, and do not specify where or when, or in what context, the action takes place. In general, molecular functions are activities that can be performed by individual gene products, but some activities are performed by complexes of gene products that work together. For example, a broad functional term is catalytic activity, or transporter activity, or binding, and an example of a more specific term is adenylate cyclase activity or Toll receptor binding (figure 1.4).


biological process
molecular function
metabolic process
cellular process
single-organism process
Is a
Part of
Has part
Regulates
Positively regulates
Negatively regulates
Occurs in
A
B
catalytic activity
nitrogen compound metabolic process
biosynthetic process
cellular metabolic process
single-organism metabolic process
organic substance metabolic process
lyase activity
cellular biosynthetic process
organonitrogen compound metabolic process
cellular aromatic compound metabolic
heterocycle metabolic process
cellular nitrogen compound metabolic
organic cyclic compound metabolic process
organic substance biosynthetic process
primary metabolic process
single-organism cellular process
phosphorus metabolic process
carbohydrate derivative metabolic process
organonitrogen compound biosynthetic process
aromatic compound biosynthetic process
heterocycle biosynthetic process
cellular nitrogen compound biosynthetic
organic cyclic compound biosynthetic process
purine-containing compound metabolic process
nucleobase-containing compound metabolic
small molecule metabolic process
phosphate-containing compound metabolic
organophosphate metabolic process
single-organism biosynthetic process
nucleobase-containing small molecule metabolic
organophosphate biosynthetic process
ribose phosphate metabolic process
carbohydrate derivative biosynthetic process
nucleobase-containing compound biosynthetic
nucleoside phosphate metabolic process
cyclase activity
purine-containing compound biosynthetic process
nucleoside phosphate biosynthetic process
nucleotide metabolic process
ribose phosphate biosynthetic process
phosphorus-oxygen lyase activity
cyclic nucleotide metabolic process
nucleotide biosynthetic process
purine nucleotide metabolic process
ribonucleotide metabolic process
cyclic nucleotide biosynthetic process
purine nucleotide biosynthetic process
purine ribonucleotide metabolic process
ribonucleotide biosynthetic process
cyclic purine nucleotide metabolic process
cAMP metabolic process
purine ribonucleotide biosynthetic process
cAMP biosynthetic process
adenylate cyclase activity

QuickGO - http://www.ebi.ac.uk/QuickGO

Figure 1.4: GO annotation terms of the Molecular Function domain for Catalytic Activity (gray) and Adenylate Cyclase Activity (gray) terms. Colors of the links establish the types of relationships between two categories.

To confuse a gene product name with its molecular function is easy. For this reason many molecular functions are annotated with the word "activity".

1.2.2 Scope of the GO

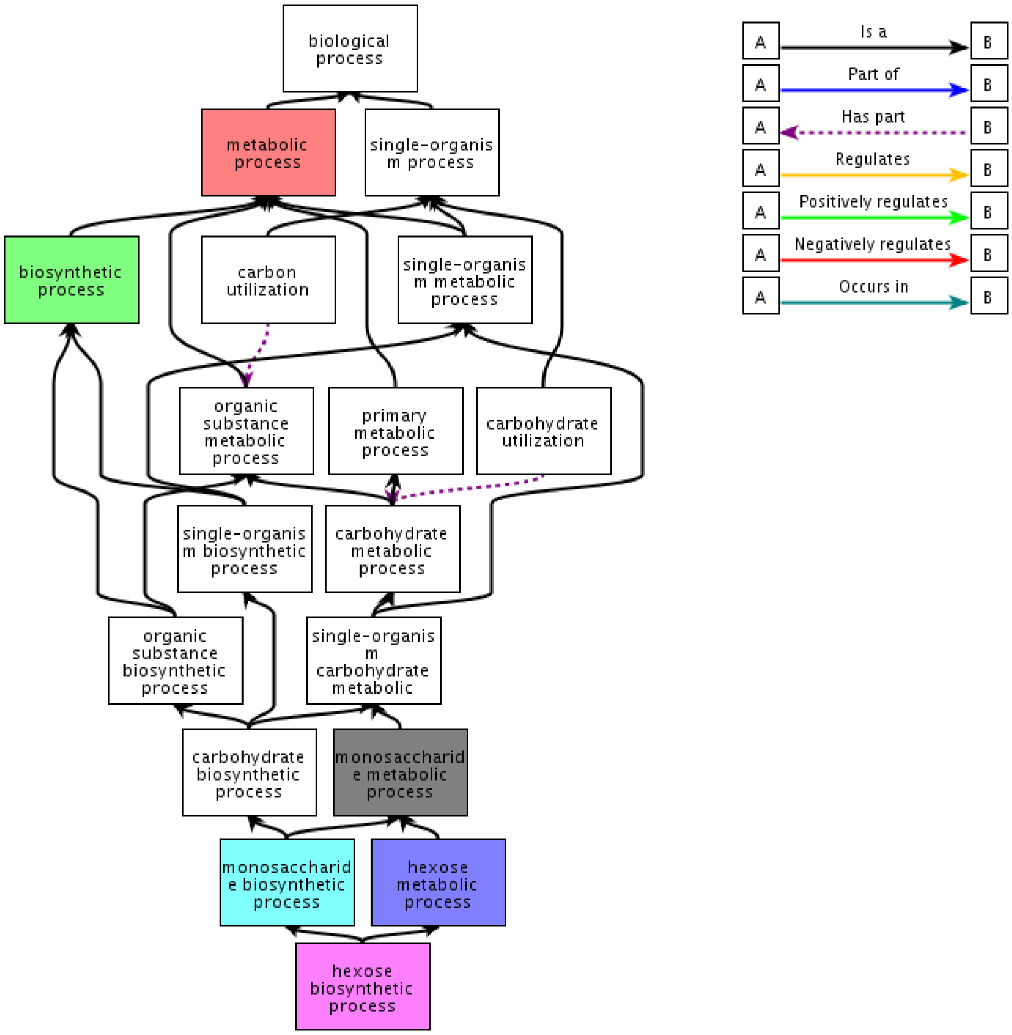
GO allows us to annotate genes and their products with a limited set of properties. For example, GO does not allow us to describe genes in terms of which cells or tissues they are expressed in, which developmental stages they are expressed at, or their involvement in disease. GO describes how gene products behave in a cellular context. However, GO is not a database of gene sequences, nor a catalog of gene products. There are other ontologies that are being developed for these purposes (e.g. the Open Biomedical Ontologies ([158])). Thus, it is important to understand that there are some areas that are outside the scope of GO. The domains covered by GO were described in previous section 1.2.1, and the following list of items shows what is not covered in the scope of the GO:

- Gene products (e.g. cytochrome complex[2] is not in the ontologies, but attributes of it, such as oxidoreductase activity, are).
- Processes, functions or components that are unique to mutants or diseases (e.g. oncogenesis is not a valid GO term because causing cancer is not the normal function of any gene).
- Attributes of sequence such as intron/exon parameters.
- Protein domains or structural features.
- Protein-protein interactions.
- Environment, evolution and expression.
- Anatomical or histological features above the level of cellular components, including cell types.

1.2.3 The Structure of the GO

Each ontology domain (i.e. MF, BP or CC) consists of a high number of terms or categories hierarchically related from least (top) to most (bottom) specialized characteristics, but unlike a hierarchy, a term may have more than one parent term. For example, figure 1.5 shows the relationships between the biological process term hexose biosynthetic process (violet) and its ancestors. This GO term has two parents, hexose metabolic process (blue) and monosaccharide biosynthetic process (cyan). This is because biosynthetic process (green) is a type of metabolic process (pink) and a hexose (blue) is a type of monosaccharide (gray).

[2] The cytochrome complex (cyt c) is a small heme protein found loosely associated with the inner membrane of the mitochondrion.


biological process
metabolic process
single-organism process
biosynthetic process
carbon utilization
single-organism metabolic process
organic substance metabolic process
primary metabolic process
carbohydrate utilization
single-organism biosynthetic process
carbohydrate metabolic process
organic substance biosynthetic process
single-organism carbohydrate metabolic
carbohydrate biosynthetic process
monosaccharide metabolic process
monosaccharide biosynthetic process
hexose metabolic process
hexose biosynthetic process
A Is a B
A Part of B
A Has part B
A Regulates B
A Positively regulates B
A Negatively regulates B
A Occurs in B

QuickGO - http://www.ebi.ac.uk/QuickGO

Figure 1.5: Relationships between a GO term and its GO term parents. Colors of the links establish the type of relationship between two categories.

Thus, the structure of GO can be described in terms of a graph, where each GO term is a node, and the relationships between nodes are links. Theses relationships are directed (e.g. a mitochondrion *is an* organelle, but an organelle is not a mitochondrion) and the graph is acyclic, that is, cycles are not allowed in the graph. Therefore, MF, BP and CC ontologies are indeed *Direct Acyclic Graphs* (DAG) and *graph theory* ([17], [162], [43], [155]) is clearly one possible, although not yet generalized, approach for their study.

Most genes are annotated in one or more categories. Annotations are made

as specific as possible. As a consequence, a gene is not only associated with its annotations but also with all the less specific terms associated with them. Furthermore, a given gene product may represent one or more molecular functions, be used in one or more biological processes and appear in one or more cellular components. For example, gene TP53 in humans encodes protein p53. This protein is essential because it regulates the cell cycle and functions as a tumor suppressor, preventing cancer. In its anti-cancer role, p53 works through several mechanisms, which are annotated in terms of the three domains of GO. Figure 1.6 describes some mechanisms and behaviors of p53 in terms of MF, BP and CC DAGs.
Together this configures a graph of terms for each gene included in the bigger graph, which is the Gene Ontology.

1.3 From Biological to Statistical Significance

To contribute to the biological interpretation from the point of view of mathematical and statistical methods, bioinformatics has focused on two major types of strategies based on the study of the Gene Ontology, among others.

These approaches are the study of *semantic similarity measures* and *enrichment analyses*. The following subsections describe both strategies.

1.3.1 Philosophy of Semantic Similarity Measures

In section 1.2.3 it was mentioned that a possible approach to navigate the relationships between GO term is based on the graph theory. To be clear, in order to assess the degree of relationship between two nodes in a graph, graph theory allows us to use the concept of distance. Thus, applying this method to biological ontologies, in particular to the GO, it is possible to measure how far or close two specific terms are housed in the topological structure of the GO. However, the concept of distance is difficult to digest when we are talking about biological interpretation. For instance, in figure 1.6, the hexose biosynthetic process is a hexose metabolic process, and it is a biosynthetic process, too. So, if we look at the distance between these GO terms, notice that the hexose biosynthetic process is close to the hexose metabolic process, but the biosynthetic process is far from the hexose biosynthetic process. For this reason, instead of using the distance measure between two terms, it is more appropriate to use some sort of function that measures the similarity

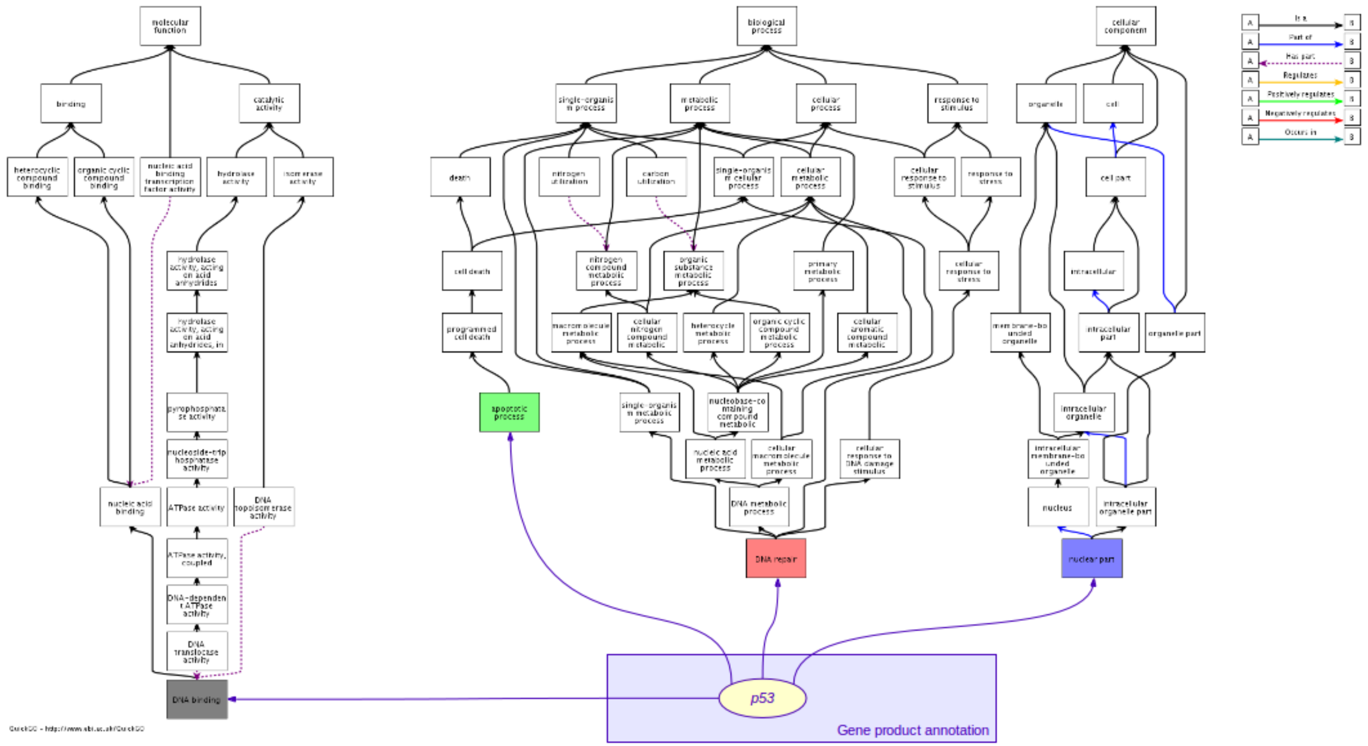

molecular function
binding
catalytic activity
heterocyclic compound binding
organic cyclic compound binding
nucleic acid binding transcription factor activity
hydrolase activity
isomerase activity
hydrolase activity, acting on acid anhydrides
hydrolase activity, acting on acid anhydrides, in
pyrophosphatase activity
nucleoside-triphosphatase activity
nucleic acid binding
ATPase activity
DNA topoisomerase activity
ATPase activity, coupled
DNA-dependent ATPase activity
DNA translocase activity
DNA binding
QuickGO - http://www.ebi.ac.uk/QuickGO
biological process
single-organism process
metabolic process
cellular process
response to stimulus
death
nitrogen utilization
carbon utilization
single-organism cellular process
cellular metabolic process
cellular response to stimulus
response to stress
cell death
nitrogen compound metabolic process
organic substance metabolic process
primary metabolic process
cellular response to stress
programmed cell death
macromolecule metabolic process
cellular nitrogen compound metabolic
heterocycle metabolic process
organic cyclic compound metabolic process
cellular aromatic compound metabolic
apoptotic process
single-organism metabolic process
nucleobase-containing compound metabolic
nucleic acid metabolic process
cellular macromolecule metabolic process
cellular response to DNA damage stimulus
DNA metabolic process
DNA repair
cellular component
organelle
cell
cell part
intracellular
membrane-bounded organelle
intracellular part
organelle part
intracellular organelle
intracellular membrane-bounded organelle
nucleus
intracellular organelle part
nuclear part
p53
Gene product annotation
A
Is a
B
Part of
Has part
Regulates
Positively regulates
Negatively regulates
Occurs in

Figure 1.6: Mechanisms and behaves of p53 in terms of MF, BP and CC DAGs. p53 has been identified in the nuclear part (blue) that is a cellular component. It activates the biological process of DNA repair (pink) proteins when DNA has sustained damage, or initiates apoptosis process (green), that is also a biological process, if DNA damage proves to be irreparable. Activated p53 binds DNA (gray) and activates expression of several genes including some microRNA miR-34a [113].

between such terms. In fact, in statistics, similarity measures are well known tools, and usually a similarity measure is defined as the inverse of a distance. In other words, relationships between annotations in the context of biological ontologies may be quantified by similarity measures, and such measures are functions that quantify the similarity between pairs of semantic terms. That is, the more similar two concepts are, the greater their similarity must be, and the smaller the distance between.

1.3.1.1 Semantic Similarity Methods

In terms of biological interpretation being able to measure the relationship between two terms it is important. However, in itself it does not go any further. A researcher needs to be able to "read" what, how and why a phenomenon in which he/she is interested occurs. The advantage of ontologies in front of a simple database is that they allow building "phrases" with a subject (i.e. terms), a verb (i.e. the type of relationship) and a predicate (i.e. restrictions of the relationship). In linguistic analysis of a set of terms with an ontological structure, a number of metrics to compute the level of similarity of the syntactic content of such terms have been defined. These metrics are called *semantic similarity measures*. The idea of distance between the terms is based on the affinity of their meaning as opposed to similarity that can be calculated regarding their syntactical representation (i.e. their ontological structure). Therefore, it is not surprising that these metrics have been widely accepted by statisticians and bioinformaticians in the study of biological ontologies with the aim of organizing and summarizing biological information. For example, in figure 1.6, there is no direct relationship between the terms monosaccharide biosynthetic process and hexose metabolic process, but both terms are a metabolic process, and they are also a monosaccharide metabolic process. Note that a similarity measure can tell us that a metabolic process has a lesser level of relationship with both specialized terms than a monosaccharide metabolic process has. However, it does not take into account the discourse of these terms and relationships at the same time. That is, a metabolic process is a term including many different concepts (e.g. primary metabolic process, organic substance biosynthetic process,...), but leaving aside the distance between this term and its offspring, syntactically it does not describe most of them. For example, just by looking at the meaning of hexose metabolic process it is quite clear that this concept describes a specialization of a metabolic process, however, this does not happen with the monosaccharide biosynthetic process. A semantic similarity has the capability to take into account the discourse of annotations.

Large lists of semantic similarity measures have been suggested and they have been classified in many different ways ([72]). For example, a classification based on cognitive models ([72]) proposes four major categories: structural measures, feature-model measures, information theory measures, and hybrid measures. But, probably, the most accepted organization is according to the elements of the graph ([122]), where measures are classified into three categories: edge-based approach, node-based approach, and hybrid approach. In fact, often, edge-based measures refer to structural measures, node-based measures are unfolded into feature-model and information theory measures, and hybrid measures refer to those measures which mix several criteria. Figure 1.7 shows a schema of the main approaches used for computing semantic similarity measures based on these two classifications.

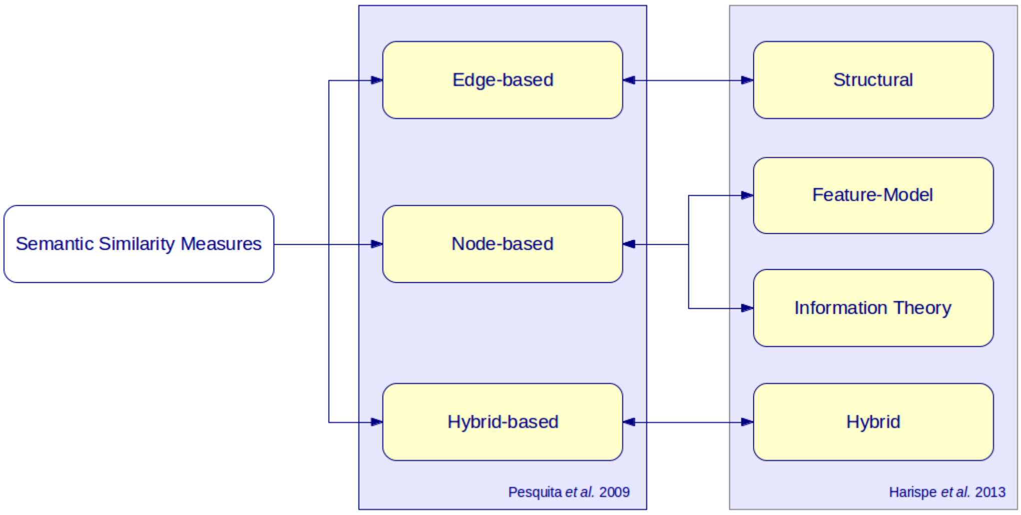


Figure 1.7: Schema of two classifications of semantic similarity measures

The following subsections describe the main idea of each type of semantic similarity measure according to its corresponding category.

1.3.1.2 The Edge-Based Approach

It consists of measures focused on relationships between concepts. These measures try to find out how similar two concepts are, by counting the number of edges that exist in the graph path between the corresponding nodes. For instance, a common procedure is to compute a distance by selecting either the shortest-path [127] or the average of all paths, when there is more than one path linking the two nodes, and then converting this distance into a similarity measure.

1.3.1.3 The Node-Based Approach

They are measures focused on the properties of the concepts being compared. These measures examine how similar the concepts are by taking into account properties that are attributable to the concepts themselves, their ancestors and/or their descendants. For example, one of the most widely used techniques is based on the *Information Content* (IC) [128]. This measure calculates the amount of information that a concept conveys, allowing to interpret how informative and specific such a concept is. IC can be computed in different ways. For instance, one of these methods is to count the number of offspring that a particular concept has. So, when two concepts in the ontology structure are being compared, the IC of common ancestors are used to give a measure of semantic similarity. To do this, there are two main approaches: (i) the *Most Informative Common Ancestor* (MICA) and (ii) the *Disjoint Common Ancestors* (DCA) [122].

1.3.1.4 The Hybrid-Based Approach

This approach considers measures that take advantage of both edge-based and node-based measures [72]. That is, hybrid-base measures take into account different types of properties like the depth of the concept in the ontology, the number of children associated with the term, the IC, etc.

1.3.2 Philosophy of Enrichment Analysis

In section 1.1 it was explained that some efforts have been made to define the Biological Significance concept and it has no clear relation with the Statistical Significance. However, what is true is that the Biological Significance encompasses different aspects of biological knowledge, as well as the methods used to annotate a set of objects under study with information stored in different sources of such a knowledge. For example, in section 1.2.1 it was seen that relevant aspects of biological knowledge are functional annotations, that consist of assigning biological and biochemical functions to elements under study, or involve regulations and interactions, or expression, etc.; biological processes of living organisms made up of any number of chemical reactions or other events that result in a transformation; and cellular components referring to the unique, highly organized substances of which cells and organisms are composed. With regard to the methods, possibly the most well-known and widely used approach to obtaining Biological Significance in terms of Statistical Significance is a collection of methods called *Enrichment Analysis* ([79]).

The principal basis behind enrichment analysis is that a function, a process or a component is not normal in a study by itself. The reason for this is that a set of features (i.e., genes, proteins or metabolites) that co-operate together should have a higher probability of being selected by the high-throughput technology that has been used in the study. That is, these features should be potentially relevant or enriched. Therefore, instead of looking at the biological meaning of a single gene, for example, the objective of the enrichment analysis is to consider a relevant gene group-based analysis, and so increase the likelihood of researchers identifying the most appropriate biological information for the phenomena under study.

Notice that annotation spaces, such as GO, where biological information knowledge is described as gene-to-annotation, are very suitable for enrichment analysis based on high-throughput data.

1.3.2.1 Enrichment Analysis Tools

During the last decade there have been many methods developed with the aim of studying biological meaning based on the enrichment analysis ([9]). Different authors consider many related methods and applications that apply the “same” idea ([47]) in different ways. However, what they have in common is that most of the tools devoted to this task generally work in two systematic steps. They consider a list of interesting genes from a population (*aka* universe or reference), resulting from a high-throughput experiment. The first step consists in mapping each gene of interest to all the annotation terms that are associated with it, and the second step is quantifying the enrichment of genes annotated in each category by comparing the proportion of genes of interest that were assigned to such a category versus the proportion of genes from the universe that were assigned to the same category (figure 1.8).

Tools for enrichment analysis have been classified into three major categories ([79]): *Singular Enrichment Analysis* (SEA) ([89], [45]), *Modular (or concurrent) Enrichment Analysis* (MEA) ([79], [80]), and *Gene Set Enrichment Analysis* (GSEA) ([146]). The following subsections describe the principles of each approach.

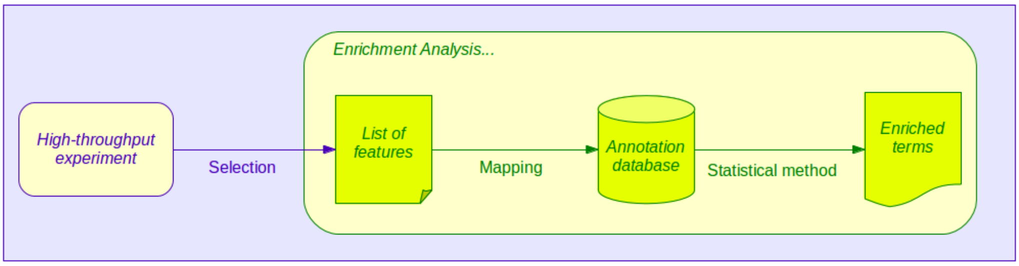


Figure 1.8: Outline of steps that follow most of tools to perform an enrichment analysis.

1.3.2.2 Singular Enrichment Analysis

SEA is the most widely used approach for enrichment analysis ([89], [45]). Briefly, a selected gene list is used to query different annotation terms one by one. That is, SEA consists in taking a selected list of interesting genes (e.g. it considers a list of differentially expressed genes selected from a comparison between experimental and control conditions by applying a statistical criterion), and it then performs a statistical test to survey the enrichment of each annotation term independently, iteratively, and one-by-one. After that, terms whose p-values are lower than a threshold of significance are reported in a table ordered by the enrichment p-value (i.e. the probability of the number of genes in the list that have fallen into a specific term compared with random chance). Statistical methods commonly used are Hypergeometric distribution, Fisher's Exact Test, Chi-square, or Binomial probability.

1.3.2.3 Modular Enrichment Analysis

Annotation terms in a database are highly redundant, and also have strong interrelationships regarding the same biological phenomenon. Thus, considering such relationships are closer to the biological reality. MEA [79] and [80] inherits the idea of SEA, but adds network discovery methods by considering relationships between terms. That is, a selected gene list is used to test multiple terms at once. The advantage of this approach is that relationships between each pair of terms may contain unique biological meaning for a given experiment that is not held by single terms. The idea behind this approach is to re-organize complex co-occurrences retrieved from multiple heterogeneous annotations (e.g. GO terms, protein domains and KEGG pathways) into gene classes with a measure of agreement, such as Kappa statistic, combined with the computation of SEA enrichment p-values.

1.3.2.4 Gene Set Enrichment Analysis

GSEA ([146]) is completely different from SEA and MEA. It takes into account the magnitude of measure differences between conditions for each gene resulting from the high-throughput experiment. It asks the question of whether the measures of the gene set of interest show significant differences between conditions. GSEA relies on more than 10000 pre-defined gene sets (currently, when writing this thesis, there are 10295 gene sets) collected from different databases (such as GO or KEGG databases) and computational studies, which are stored in a database called *Molecular Signatures Database* (MSigDB) ([146]). The gene sets in MSigDB are divided into the following seven major collections.

- *c1: positional gene sets* for each human chromosome and cytogenetic band.
- *c2: curated gene sets* from online pathway databases, publications in PubMed, and knowledge of domain experts.
- *c3: motif gene sets* based on conserved cis-regulatory motifs from a comparative analysis of the human, mouse, rat, and dog genomes.
- *c4: computational gene sets* defined by mining large collections of cancer-oriented microarray data.
- *c5: GO gene sets* consist of genes annotated by the same GO terms.
- *c6: oncogenic signatures* defined directly from microarray gene expression data from cancer gene perturbations.
- *c7: immunologic signatures* defined directly from microarray gene expression data from immunologic studies.

This organization allows us to restrict the search to specific groups of genes that have traits associated with the interest pursued by the researchers.

The basic idea of the statistical method that is behind GSEA consists in testing for enrichment of some gene set among genes from the high-throughput experiment. Specifically, GSEA ranks the whole list of genes according to the correlation of the profiling measures generated in the high-throughput experiment with the phenotype. This is performed in order to calculate the fraction of genes in a gene set weighted by their correlation ("hits"), and the fraction of genes not present in the same gene set ("misses") up to a given position in the whole list of genes. So, these fractions are used to calculate the enrichment score of the gene set ES(S), which is the maximum deviation from zero. Usually, to compute the ES, the Kolmogorov-Smirnov statistic is applied, but alternative parametric statistical approaches such as z-score, t-test, or permutation analysis have been purposed.

Chapter 2

Hypothesis, Objectives and Outline of the Thesis

2.1 Hypotheses

Based on high-throughput data generated in omic experiments, different approaches to studying biological information associated with these data, and stored in heterogeneous annotation databases of bioinformatic sources, have been proposed. One of these types of sources is biological ontologies, and probably the most widely accepted and used is the Gene Ontology (GO). In order to deal with the annotation terms of the GO, two main strategies have been adopted. On the one hand, semantic similarity measures that allow the summarization and comparison of such terms and, on the other hand, enrichment analyses that allow the extraction and attribution of biological meaning to a large list of features resulting from a particular omic experiment. However, both strategies have some weaknesses and fundamental issues not yet studied. Specifically:

- Regarding the semantic similarities, large lists of measures have been proposed ([128], [98], [84], [82]), and they have been organized and classified in different ways ([122], [72]). However, current state-of-the-art of semantic similarity measures:

  1. Has no mathematical proofs to demonstrate that these types of similarities are indeed similarity measures, understanding these as reverse complementary of metric distances.
  2. In addition, there is also a weak linkage between semantic similarities from the node-based approach and the edge-based approach, when they are applied to the specific case of the Gene Ontology.

- Concerning GO tools for enrichment analysis, many different methods and software have also been proposed ([47]). Some efforts to classify these tools based on the type of method applied for the enrichment of GO terms have been performed ([79]). However,

1. When a researcher is looking for software to perform an enrichment analysis, it is highly possible that it is lost, or at least that he/she does not find the most suitable tool for his/her needs, due to the large number of existing GO tools, even having classified the GO tools based on the type of enrichment that they carry out.
2. The speed of development of new methods and tools for the enrichment analysis, as well as the improvement of existing applications, by the scientific community, is really considerable. In this regard, neither has any comprehensive monitoring of tools been conducted to see how their capabilities evolve, nor has a general strategy been organized for the development of new applications.

In order to shed light on these issues, the main and specific objectives that this thesis will address are presented in the following sections.

2.2 Objectives

The context of this thesis is focused on the methods and tools that are used to attribute biological meaning based on data generated with high-throughput technologies in omic experiments, with special emphasis on lists of differentially expressed genes generated with microarrays in the field of transcriptomics and the discourse of the Gene Ontology.

2.2.1 Main Objectives

This thesis has two main objectives:

1. The study of two types of semantic similarity measures for exploring GO categories.

2. The classification and study of GO Tools for traditional enrichment analysis.

2.2.2 Specific Objectives

In order to accomplish the main objectives, two major lists of specific objectives are considered.

2.2.2.1 Specific Objectives Associated with the Study of Semantic Similarity Measures

The specific objectives associated with the study of semantic similarity measures are:

1. Proofs that both approaches are related to the concept of metric distance.

2. Development of an R package for computing semantic similarity measures between terms of any ontology and compare semantic similarity profiles.

2.2.2.2 Specific Objectives Associated with the Classification and Study of GO Tools for Enrichment Analysis

The specific objectives associated with the study of GO Tools for traditional enrichment analysis are:

1. Definition of a list of functionalities that allows us to classify GO tools for enrichment analysis.

2. Classification of existing GO tools for enrichment analysis based on the list of functionalities and according to their capabilities.

3. Development of a web-based application intended to select and compare the GO tools that are best suited for the needs of a user.

4. Study of the GO tools evolution in order to characterize the existence of representative models.

5. Construction of an ontology for organizing a vocabulary devoted to developing new GO tools.

2.3 Outline

The thesis consists of eight chapters divided into four parts. After this introduction and problem formulation, Part 2 surveys major lines of research on the study of two types of measures regarding the semantic similarity approach for exploring the relationship between pairs of GO categories. This part is organized into three chapters. The first chapter introduces and describes the materials and methods used. The second chapter presents the results. The third chapter discusses the contributions. Part 3 deals with the classification and study of Gene Ontology Tools for traditional enrichment analysis. In this part, the research is also organized into three chapters, similarly to the second part. Finally, Part 4 presents the conclusions. Additionally, in the appendixes, published results, and some extra information used in carrying out the research are presented. Figure 2.1 shows the organization of the different parts of the thesis and the relationship between chapters.

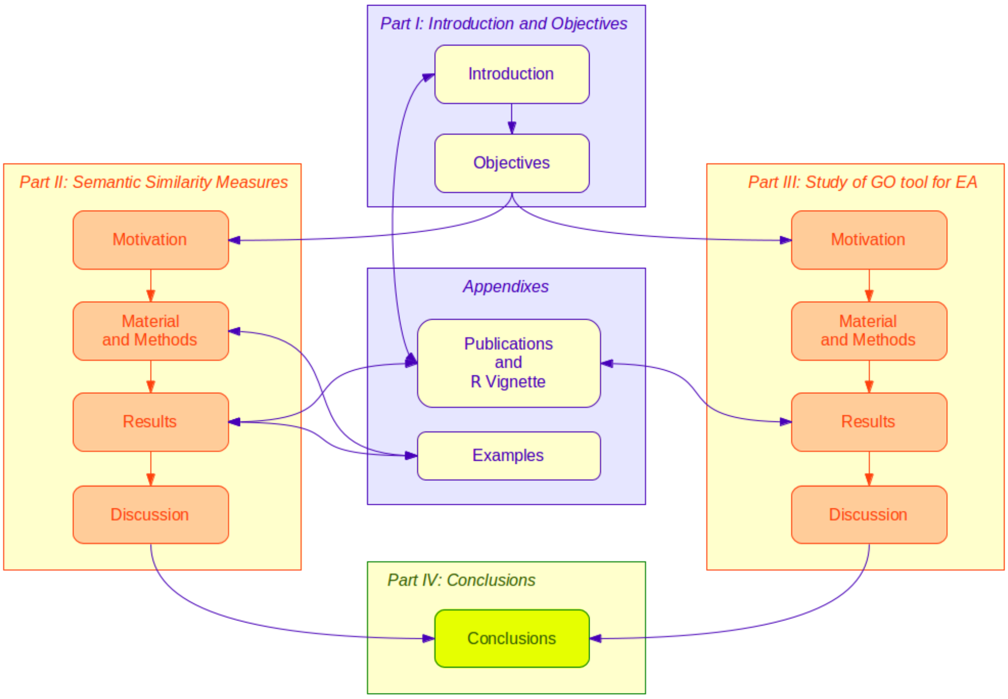


Figure 2.1: Organization of the different parts of the thesis and the relationship between chapters.

Part II

Study of Two Semantic Similarity Measures for Exploring GO Categories

The three cornerstones of experimental science are observation, classification and comparison. A very good example of this philosophy is the study of molecular biology. Usually, biological knowledge is stored in databases, whose records consist of a set of fields describing the experiment, the methodologies and the results, often written in natural language. When researchers discover something new, they try to infer new biological knowledge according to the "degree of similarity" between the new observed entity and the previous classified knowledge stored in an specific database. However, looking for "something similar" is a concept that is sometimes hardly definable. For instance, when comparing a gene sequence with genome sequences stored in a database, applying an alignment algorithm is a relatively easy thing to do, while comparing the annotations describing biological processes or functional characteristics is not so easy to perform. Moreover, when "looking for something similar" is well established, searching for matches in a database might not be viable in computing time because of the huge quantity of information stored.

The advent of the omics era has had a deep impact on molecular biology knowledge. Instead of dealing with experiments focused on a single feature level, omic experiments have allowed us to deal with a large-scale features level. The completion of several genomes has generated enormous quantities of sequence data, and with the continuous development of high-throughput technologies, the amount of functional data has increased dramatically. Thus, it has become crucial to develop strategies that help researchers to annotate and classify all the observed data, and then permit comparison with new discoveries afterwards. Different approaches such as databases with cross-linked annotations, schemes, or taxonomies have been proposed. But, probably the most widely used way to classify entities and annotate concepts has been ontologies. Three significant examples of these methodologies to organize and store large quantities of data are: Sequence Ontology (SO), which provides a structured controlled vocabulary for sequence annotation [50], PRotein Ontology (PRO), which is designed to describe the relationships of proteins [114], and Gene Ontology (GO), whose objective is the standardization of the representation of gene and gene product attributes across species and databases [148].

Informally, an *ontology* is a way of annotating concepts in a certain domain that allows comparison between entities through their associated concepts, and which would not otherwise be comparable. For instance, very often researchers are interested in comparing the pathways associated with two lists of genes, whose expression profiles have shown significant differences. If the associated gene products are annotated in the GO, then these two lists of genes might be compared through the terms in which the associated gene products were annotated. Many methods for performing these types of analyses have been proposed. A usual way of comparing two concepts within an ontology is by looking at common terms, but a more sophisticated way to proceed relies on *similarity measures* [15].

This part of the thesis is devoted to answer the first main objective in 2.2.1. That is, to studying a couple of measures about the semantic similarity approach for exploring GO categories. To carry out such a study, the specific objectives stated in section 2.2.2.1, are answered one by one. Thus, in chapter 3 material and methods associated with specific objectives are introduced or described, in chapter 4 results associated with specific objectives are presented, and finally, in chapter 5 results are discussed.

Chapter 3

Material and Methods

3.1 Main Concepts of Graph Theory

The Gene Ontology Consortium states in its web that “GO terms are organized in structures called *Directed Acyclic Graphs* (or DAGs), which differ from hierarchies in that a child, or a more specialized term, can have many parents, or less specialized terms” ([148]). In fact, very often the Gene Ontology is defined as a DAG.

Given a list of genes, their associated GO terms form a subgroup of the ontology which is called the *induced subgraph*. That is, given one or more gene products, it is possible to recover their associated terms and their ancestors in the biological processes, molecular functions or cellular components (see section 1.2.1). These ancestors constitute an unstructured list of GO terms whose associated annotations might be useful for certain purposes. However, this list of GO terms does not contain any information about the relation between the GO terms associated with the query genes list. This kind of information is contained in the subgraph induced by the genes. So, it is hardly surprising that an alternative way to perform “GO Analysis” is to consider the induced subgraphs as the starting point of the analysis.

There are different approaches that use the graph theory ([43], [17]) as the basis for the ontological analysis. For instance, in the case of the GO, a theoretical approach that has been widely explored is the notion of defining distances between two genes in terms of the similarity of their GO annotations. In this sense, we will see in section 3.5.3, there are different strategies that have been proposed for defining distances, or similarities between genes based on GO annotations. Two different approaches have been explored by Lord *et al.* ([98]) and Joslyn *et al.* ([85]). The former makes use of the *Information Content* in the GO as the basis for assigning similarity between terms. And the latter adopt a different strategy that relies on the inherent structure of a general graph to define distances, pseudo-distances indeed, between the GO terms.

In any case, if one wishes to either work with semantic similarities or simply be able to compute any kinds of distances between nodes of a graph, it is necessary to establish the main concepts of the graph theory. Therefore, in the following subsections, some graph theory concepts are introduced that will be used later to better understand the structure of the Gene Ontology.

3.1.1 Basic Graph Concepts

Many problems of daily life can be represented by a diagram consisting of a set of concepts joined by certain relations. A mathematical abstraction to solve this kind of situation is the idea behind a *graph*.

Definition 3.1. *A* ***graph*** *is a pair $G = (V, E)$, where V is a set called* ***vertex set*** *whose elements are called* ***vertices*** *(or* ***nodes*** *or* ***points****), and $E \subseteq \{e_{ij} = (v_i, v_j) : v_i, v_j \in V\}$ is a binary relation on V where a pair e_{ij} is called* ***edge*** *(or* ***arcs*** *or* ***lines****), $i, j \in I \subseteq \mathbb{N}$.*

The most important advantage of working with graphs is that it makes it possible to have a visual representation of the problem. For each vertex $v \in V$ a point is usually drawn in the plane, and for any two vertices $v_i, v_j \in V$ a line joining them is drawn. The use of this abstraction is a good way to translate the real life problem into a mathematical form.

Notation 3.1. *The number of nodes and edges in graph $G = (V, E)$ are denoted by $\nu(G)$ and $\varepsilon(G)$, respectively.*

There is no a unique way to draw a graph. The main idea is to realize which pairs of vertices give an edge and which of them do not.

Definition 3.2. *Let $v_i, v_i^k \in V$ be two nodes with $i \in I$, $k \in \mathbb{N}$. Let $e_{ij}, e_{ij}^k \in E$ be edges such that $e_{ij} = (v_i, v_j)$ and $e_{ij}^k = (v_i^k, v_j^k)$, $i, j \in I$, $k \in \mathbb{N}$. Then,*

1. *An edge e_{ij} is a* ***self-loop*** *if $v_i = v_j$.*

2. *Edges e_{ij}^1, e_{ij}^2 are* ***parallel edges*** *if $v_i^1 = v_i^2$ and $v_j^1 = v_j^2$.*

3. *An edge e_{ij} is* ***incident*** *on vertices v_i and v_j.*

4. *Two vertices $v_i, v_j \in V$ are* ***adjacent*** *(or* ***neighbours****) if $\exists e_{ij} \in E$.*

Definition 3.3. *A graph is called* ***simple graph*** *if it contains neither self-loops nor parallel edges. If a graph contains self-loops or parallel edges it is called* ***multigraph****.*

Examples B.1 and B.2) in appendix B show the representation of a graph and a multigraph respectively.

From now on, simple graphs are assumed, unless the opposite is indicated.

3.1.2 Subgraphs

Definition 3.4. *A graph $S = (V_S, E_S)$ is a* ***subgraph*** *from $G = (V, E)$ if $V_S \subseteq V$ and $E_S \subseteq E$. Dually, if S is a subgraph from G, then G is a* ***supergraph*** *for S.*

In other words, a subgraph from G induced by V_S is a graph $S = (V_S, E_S)$ such that E_S contains all the edges from E that exist between nodes of V_S.

Example B.3 in appendix B shows the representation of a subgraph.

3.1.3 Directed Graphs

Despite the many problems that can be solved with graph theory approaches, as mentioned in section 3.1, sometimes this cannot be done if no restrictions are introduced. For instance, let G be a graph whose edges E link two specific vertices. But, imagine that the two vertices must only be connected in one direction. That is, the vertices have to only be connected when "travelling" from the *source* vertex (origin) to the *target* vertex (terminus). So, a notion of "orientation" is required.

Definition 3.5. *A* ***directed graph*** *D (or* ***digraph****) is an ordered pair (V, E) of disjoint sets of vertices and edges, and an incidence function ψ that assigns for each edge an ordered pair of vertices,*

$$\begin{array}{rccl} \psi : & E & \longrightarrow & V \times V \\ & e & \longmapsto & \psi(e) = (v_i, v_j). \end{array} \tag{3.1}$$

We say that edge e ***joins from*** *v_i* ***to*** *v_j, where v_i is the* ***initial node*** *(or* ***source****), and v_j is the* ***terminal node*** *(or* ***target****).*
Edges in a digraph are also called ***arcs****.*

A graph whose edges connect nodes in both ways is called an *undirected graph* (*aka symmetric graph*). Digraphs are usually drawn with arrows to indicate the arc directions.

The *incidence function* ψ must be taken into account when working with digraphs, because it plays a central role. That is, given a subgraph S induced by V_S from a digraph D, the incidence function ψ_S is the restriction of ψ for the digraph D to E_S.

Definition 3.6. *A digraph D is an **orientation** of a graph G if*

1. *The sets of vertices and arcs from D, and the sets of vertices and edges from G are the same.*
2. *Given an incidence function ψ, then $\psi(e_{ij}) = (v_i, v_j) = \psi_{ij}$ for every edge $e_{ij} = (v_i, v_j)$.*

Thus, given a graph G, an oriented graph D can be obtained by joining each edge from one of the ends to the other. Conversely, for each digraph D, a graph G can be obtained by assigning for every arc from D an edge from G with the same ends. In that case, G will be the *underlying graph* of D.

Example B.4 in appendix B shows the representation of a digraph.

3.1.4 Paths and Connection

Definition 3.7. *A **path** is a graph $P = (V_P, E_P)$ such that*

$$V_P = \{v_0, v_1, \ldots, v_k\} \quad , \quad E_P = \{e_{01}, e_{12}, \ldots, e_{(k-1)k}\}$$

where, $v_i \neq v_j$, $\forall i \neq j$.

However, very often, there is a misuse of the path concept. Given a graph G, the *path between two nodes* is usually understood as a natural sequence of the nodes that there are between them. Formally,

Definition 3.8. *Let $G = (V, E)$ be a graph. A **path** P **from** v_0 **to** v_k is a sequence of nodes $P = (v_0, v_1, \ldots, v_k)$ in G such that the target node of each edge is the source node of the next edge in the sequence where v_0 is the **origin** node, v_k is the **terminus** (both commonly called the **ends** of the path), and $v_1, v_2, \ldots, v_{k-1}$ are the **internal** nodes.*

Definition 3.9. *Let $G = (V, E)$ be a graph. If there is path P with origin v_0 and terminus v_k, then we say v_k **is reachable from** v_0.*

So, the first way to know how far apart two nodes are is to count the number of edges that exist between them.

Definition 3.10. *Let P be a path from v_0 to v_k in a graph $G = (V, E)$. The* ***length*** *is the number $k \in \mathbb{N} \cup \{0\}$ of edges in the path P and is denoted by P^k.*

Example B.5 in appendix B illustrates the ideas of path and length in a digraph.

Definition 3.11. *Let $G = (V, E)$ be a graph. Two vertices $v_i, v_j \in G$ are* ***connected*** *if there is a path P in G that joins them.*

The connection could be interpreted as an *equivalence relation* of a set of nodes V ([51]). This suggests that there is a partition $V_1, V_2, \ldots, V_p \in V$ where $V_k \neq \emptyset$, $\forall k = 1, 2, ..., p$ such that v_i and v_j are connected $\Leftrightarrow v_i, v_j \in V_r$.

3.1.5 DAG and Rooted DAG

Definition 3.12. *Let $P = (v_0, v_1, \ldots, v_k)$ be a path in a graph $G = (V, E)$. Then, we define*

1. *a* ***cycle*** *as a path such that $v_0 = v_k$,*
2. *an* ***acyclic graph*** *as a graph with no cycles, and*
3. *a* ***directed acyclic graph (DAG)*** *as a digraph with no cycles.*

Example B.6 in appendix B shows the representation of a DAG.

There is a kind of DAG that contains an special node which is the "father" of the remaining nodes in the DAG, however, it is an "orphan" node. Such a vertex is known as the **root** (*aka* **top**) node.

Definition 3.13. *We say* ***rooted DAG*** *a DAG $D = (V, E)$ such that it has a root node.*

3.1.6 Matrices and Graphs

Visual representations of graphs are powerful tools to quickly observe how a set of nodes are related. However, from an analytical point of view and in order to both explore and quantify such relationships it is much better to work with alternative mathematical objects such as matrices.

Thus, graph structures are usually “converted” into associated matrix forms.

In this subsection basic matrix concepts associated with graphs are introduced.

Definition 3.14. *Let $G = (V, E)$ be a graph. We define the **adjacency matrix** of G as a matrix $\mathbf{A}_G = (a_{ij})_{n \times n}$ such that $a_{ij} \in \mathbb{N} \cup \{0\}$ is the number of edges between v_i and v_j, and $n = \nu(V)$.*

That is, an adjacency matrix is a matrix whose elements are the number of edges between each pair of nodes. Notice that $a_{ij} = 0$ when there is no connection between nodes v_i and v_j.

If G is a simple graph, then $diag\,(\mathbf{A}_G)$ is a null vector because such a graph has no self-loops, and each a_{ij} only takes value 1 if $(v_i, v_j) \in E$ or 0 otherwise, since it has no parallel edges.

In a digraphs, as a general rule the adjacency matrix is read from rows (origin nodes) to columns (terminus nodes) to know if a vertex is reachable from another one.

Following on with that mentioned at the end of the subsection 3.1.1, adjacency matrices of simple graphs are the focus of this thesis, unless the opposite is indicated.

Usually, the number of entries with a 0 is substantially higher than entries with 1. In these cases, we say that a matrix (or a graph) is *sparse* when $\varepsilon(G) < \frac{\nu(G)(\nu(G)-1)}{2}$ and is *dense* otherwise.

Definition 3.15. *Let $G = (V, E)$ be a multigraph where $V = (v_1, v_2, \ldots, v_n)$ and $E = (e_1, e_2, \ldots, e_m)$. The **incidence matrix** of G is a matrix $\mathbf{B}_G \in \mathcal{M}_{n \times m}\left(\{0, 1, 2\}\right)$ where b_{ij} is the number of times that v_i and e_j are incident.*

Since the present chapter is focused on DAGs, it is necessary to redefine this matrix.

Definition 3.16. *Let $D = (V, E)$ be a DAG where $V = (v_1, v_2, \ldots, v_n)$ and $E = (e_1, e_2, \ldots, e_m)$. The **incidence matrix** of D is a matrix $\mathbf{B}_D =$*

$(b_{ij})_{n\times m}$ *where*

$$b_{ij} = \begin{cases} 1 & \text{if } v_i \text{ is the source node of } e_j \in E \\ -1 & \text{if } v_i \text{ is the target node of } e_j \in E \\ 0 & \text{otherwise.} \end{cases} \tag{3.2}$$

Definition 3.17. *Let $D = (V, E)$ be a DAG where $V = (v_1, v_2, \ldots, v_n)$. The **accessibility matrix** associated with a digraph D is a matrix $\mathbf{R}_D = (r_{ij})_{n\times n}$ such that*

$$r_{ij} = \begin{cases} 1 & \text{if } v_j \text{ is accessible from } v_i \\ 0 & \text{otherwise,} \end{cases} \tag{3.3}$$

This matrix can be straightforward computed by

$$\mathbf{R} = \bigoplus_{k=0}^{n-1} A^k \tag{3.4}$$

where $A^0 = Id$ and the truncated addition operator $\oplus$ is such that $a \oplus b := \min\{1, a + b\}$.

Examples B.7, B.8 and B.9 in appendix B illustrate the adjacency matrix, the incidence matrix, and the accessibility matrix of a DAG, respectively.

3.1.7 Order and Degrees

After introducing the general structures of graphs, the next step required is to establish some methods for "describing" what a graph is like. In notation 3.1 two measures were introduced devoted to quantifying how many nodes $\nu(G)$ and edges $\varepsilon(G)$ there are in a graph. However, in order to answer this, more sophisticated questions are required that go beyond these numbers and are associated with similarities. For instance, we would like to know how many edges are coming out from one specific node, or what kind of relations there are between nodes and edges in a graph.

In this subsection, some concepts to characterize a graph are introduced. Thus, a formalization of the number of nodes in a graph is the first concept required.

Definition 3.18. *Let $G = (V, E)$ be a graph. Then, we define*

1. *the **order** of the graph G as the number of vertices in G, $|G| = \nu(G)$, and*

2. *the* ***degree*** *(or* ***valency****) of a vertex $v \in V$ as the number of edges incident to it, $d(v) = |E(v)|$.*

Example B.10 in appendix B illustrates the ideas of the order of a graph and the degree of a vertex.

Definition 3.19. *Let $G = (V, E)$ be a graph, then*

1. ***Minimum degree*** *of G is*

$$\delta(G) := \min_{v_i \in V}(v_i). \tag{3.5}$$

2. ***Maximum degree*** *of G is*

$$\Delta(G) := \max_{v_i \in V}(v_i). \tag{3.6}$$

3. ***Average degree*** *of G as*

$$d(G) := \frac{1}{|G|}\sum_{i=1}^{|G|} d(v_i). \tag{3.7}$$

Thus, based on these concepts, the following inequality is clearly established,

$$\delta(G) \leq d(G) \leq \Delta(G) \tag{3.8}$$

The average degree might be understood as a global measure about how many edges are incident to each node in a graph.

Definition 3.20. *Let $D = (V, E)$ be a digraph, and let $v \in V$ be a vertex. Then, we define*

1. *the* ***in-degree*** *$d^{-}(v)$ of a vertex v as the number of arcs whose terminal node is v, and*

2. *the* ***out-degree*** *$d^{+}(v)$ of a vertex v as the number of arcs whose initial node is v.*

Example B.11 in appendix B illustrates the different degree measures of a graph.

3.2 The GO graph

The Gene Ontology is a rooted DAG. Actually, it is structured as three independent DAGs: Molecular Functions (MF), Biological Processes (BP) and Cellular Components (CC). Whatever the case, it is clear that the relationship between the terms of the GO may be studied according to the graph theory.

In the following subsections, specific concepts based on the graph theory for dealing with the GO DAG and performing "GO-based analyses" are introduced.

3.2.1 Basic Concepts of the GO graph

The most commonly used concepts, when working with the GO graph, are:

- **GO terms** (or **terms**: nodes in the GO graph.
- **Parents**: initial nodes associated with in-degrees of a GO term.
- **Descendants** (or **children**): terminal nodes associated with out-degrees of a GO term.
- **Ancestors**: all nodes belonging to the paths that exist between a specific GO term and the root node.
- **Offspring**: all nodes belonging to the paths between a specific GO term and all terminal nodes at the end of each of these paths, whose in-degrees are one and out-degrees are zero.

As mentioned in section 1.2.1, currently the GO DAG contains 40.000 GO terms approximately. Figure 3.1 shows a visualization of the complex architecture of the GO graph.

3.2.2 The Study of the GO based on the Graph Theory

The goal of *Gene Ontology-based analysis* is to facilitate an interpretation by means of the annotations that gene products may have in its database. However, *in contrast with the agreement found in different existing types of omic experiments, there is no well-defined classification of the (types of) problems where GO analysis might help to answer biological questions.* In the literature, some typical questions about tools and methods are found ([4]). Some of them are:

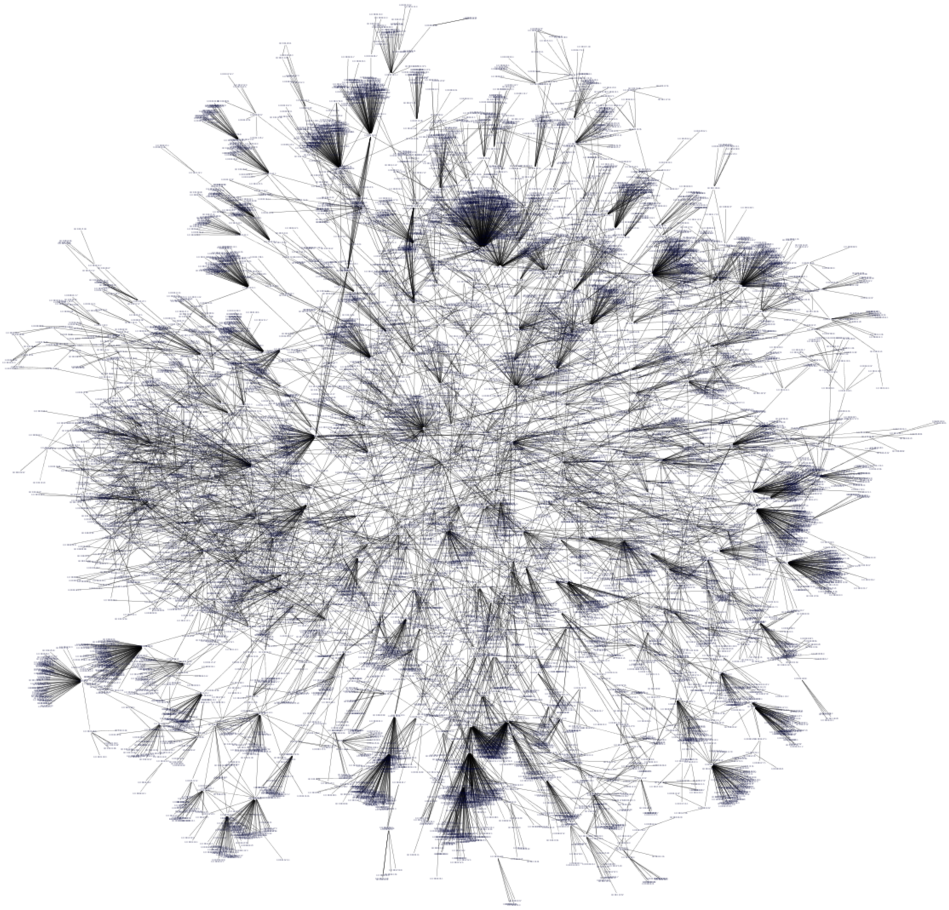


Figure 3.1: Visualization of the GO DAG. Image taken from the Cytoscape.

- What is the biological meaning of a gene list? That is, what Molecular Functions, Biological Processes or Cellular Components are these genes associated with?
- Is this set of “meanings” coherent with any biological interpretation?
- In the case of the comparison of several conditions, how should these different conditions be related based on the meaning attributed to each of them independently?
- Does a given tool enable the hierarchical structure of GO to be exploited?
- Does the analysis of a given tool enable simultaneous functional profiling for all three GO ontologies?

To answer this list of questions there is no unique approach. The strategies based on the graph theory for answering them are strongly dependent on the way used to represent or synthesize the *mapping between the gene list and the Gene Ontology*.

Given a list of genes, they are associated with certain GO terms in the GO graph. Each gene product may be associated with none, one or more than one GO term. This fact determines an induced subgraph (see section 3.1.2). That is, once the GO terms that annotate the list of genes are selected, we can retrieve all ancestors and relations between them automatically. In other words, we can retrieve an specific subgraph that is the so called induced graph.

Induced graphs may be very complex structures, especially when the list of genes producing such subgraphs is also large (i.e. hundreds or thousands). Therefore, it is important to provide a good mathematical formalization for correctly managing these structures.

3.3 Carey’s Framework

Carey, who is one of the developers of the `Bioconductor` Project ([62]), introduced a simple formalism for working with ontologies for statistical purposes ([23]). He showed how this formalism can be used for different applications. One of them is precisely the semantic similarity computation.

The following section is devoted to both introducing the main ideas of Carey's framework, and mathematically formalizing usage of the graph theory concepts that have been introduced in previous sections in order to "give biological meaning to a gene list" based on the GO.

3.3.1 Refinement of Relationships

The definition of ontology adopted by Carey ([23]) is similar to the one informally introduced at the beginning of this part of the thesis (see section II). An *ontology* provides a set of vocabulary terms covering a conceptual domain. These terms must have a definition and be placed within a structure of relationships. In order to describe relationships between terms the concept of *refinement* is presented.

Definition 3.21. *Let $\mathcal{T} = \{t_1, t_2, \ldots, t_n\}$ be a set of n terms in an ontology. We define a **relationship of 1− refinement** between two terms t_i and t_j as*

$$R(t_j, t_i) = \begin{cases} 1 & t_j \textit{ refines } t_i \textit{ and } \nexists\ t_k \in \mathcal{T} \textit{ such that } R(t_k, t_i),\ R(t_j, t_k) \\ 0 & \textit{otherwise.} \end{cases} \tag{3.9}$$

In words, t_j is a one-step refinement of t_i if there is no term t_k such that t_k is a refinement of t_i and t_j is a refinement of t_k.

Carey takes the relationship of refinement among terms to be primitive: for all $t_i \neq t_j$ it is *decidable* whether term t_i refines term t_j or not. But, in fact, 1-term refinements are an alternative way to describe the relationships between nodes in a graph. That is, in the language of nodes and edges, definition 3.21 says that given a graph having nodes t_i and t_j, there is a single directed edge leaving t_i and reaching t_j. Note that it is possible for one term to be a 1-step refinement of several other terms, as well as it being possible for one node in a graph to be joined by several edges to several distinct nodes. In other words, each term may have one or more parents.

In subsection 3.2, we have seen that the GO is a rooted DAG. That is, the GO is an ontology structured as a hierarchical DAG, whose nodes are linked by arcs with an special meaning. In terms of Carey's language, these nodes are refinements linked by relationships. In the GO there are two main types of relationships ([128]) and Lord *et al.* ([98])

- **is-a**, that establishes a relationship between a *parent* and a *child*, and

- **part-of**, meaning that a relationship exists between a *part* and the *whole*.

Note that in this sense the root term is a singular node, that is, it is refined by all other terms in the rooted DAG, but it does not refine any other term.

3.3.1.1 Refinement Matrix

In Carey's framework the adjacency matrix is reinterpreted as follows,

Definition 3.22. *Let $\mathcal{T} = \{t_1, t_2, \ldots, t_n\}$ be a set of n terms in an ontology. The matrix $\Gamma = (\Gamma_{ij})_{n\times n}$ satisfying*

$$\Gamma_{ij} = \begin{cases} R(t_i, t_j) & i > j \\ 0 & \textit{otherwise}, \end{cases} \tag{3.10}$$

is called the ***(one-step) refinement matrix****.*

Example B.12 in appendix B shows a representation of a rooted DAG with 12 terms and shows its associated refinement matrix.

The refinement matrix is not only important because it represents the graph itself, but also because many relevant measures are derived and computed from it. For instance, k-step refinements represent terms that can be accessed from other terms in k steps, and they are encoded in terms of powers of this matrix. For example, Γ^2 encodes terms that can be accessed from other terms in two steps, that is, terms separated by two edges.

Definition 3.23. *Let $\mathcal{T} = \{t_1, t_2, \ldots, t_n\}$ be a set of n terms in an ontology. Then,*

1. *the* ***depth*** *of an ontology is the zero-based length of the longest path from the most refined term to the root, and it can be computed as the number*
$$d_0 = \min k \tag{3.11}$$
such that the matrix $\Gamma^{k+1} = 0$[1]*, and*

[1] Carey defines the depth of an ontology as

$$d_0 := \min_k \max_{i,j} \Gamma^k_{ij}.$$

This definition is confusing and we have not been able to prove the equivalence with the one we have given above, which seems to agree better with the intuitive notion of depth in an ontology.

2. *the* ***depth of a term*** t_j *in an ontology is the number*

$$d_0(t_j) = \min k \tag{3.12}$$

such that the matrix $\Gamma_{1j}^{k+1} = 0$.

In section 3.1.3 a digraph was presented as a type of graph resulting from the application of an incidence function to the underlying undirected graph, where arcs e_ij join two nodes t_i and t_j from the first to the second. These arcs can be reinterpreted as a 1-step refinements. Thus, it is not difficult to see that columns in the refinement matrix are the parents and rows are the descendants. Usually, the underlying meaning in an ontology is that columns are the source nodes and rows are the target nodes. But, in Carey's language, we read that descendants (rows) are refinements of parents (columns). Note that this way of reading a refinement matrix might seem a contradiction in light of that explained in previous sections. For instance, following this criterion the arcs in figure B.6 are point backwards. However, descendants are 1-step refinements. The orientation concept takes on significant importance when the directions of edges are drawn in a digraph, and even more so when dealing with the meaning of the terms of an ontology like the GO.

3.3.1.2 Accessibility Matrix

To derive other measures from Γ the use of the truncated addition operator is required in order to define the *truncated summation*[2].

The *accessibility matrix* associated with a graph depicting an ontology is a square matrix, whose elements are equal to one if and only if a term path exists following the arcs from term t_i to term t_j, and elements are otherwise equal to zero (see section 3.1.6). This concept can be redefined in terms of Carey's notation as the matrix $\mathbf{A} = (a_{ij})_n \times n$ that can be obtained from the refinement matrix Γ by using the truncated summation as

$$\mathbf{A} = \bigoplus_{k=1}^{d_0+1} \Gamma^k \tag{3.13}$$

where d_0 is the depth of the ontology.

[2] This operation allows the formal construction of many interesting matrices. However, most of them can be described without the use of this operation so that these formalisms can be omitted without loss of understanding. Instead, the definition 3.4 can always be kept in mind as an alternative ([25]).

Example B.13 in appendix B shows the accessibility matrix associated with a DAG described in the example B.12.

3.3.2 Mapping Genes to GO

The main interest of the formalism elaborated by Carey is probably in the notion of *Object-Ontology Complex*. For instance, as we mentioned in previous sections, one of the challenges of biomedical researchers consists in analyzing a list of genes using the associated gene products annotated in the GO. These types of analyses can be performed thanks to the previously established relation between the gene products and the relationships between terms in each of the three ontologies of the GO. This fact makes this relation explicit, to perform relevant calculations, and the analysis relies on such a relation. Globally speaking, given a list of *objects* that are described by an *ontology*, we assume for simplicity purposes that each object is mapped to at least one term, with the stipulation that the mapping is made to the most refined term in each case.

Example B.14 in appendix B shows the representation of an OOC with 10 objects annotated in an ontology with 12 terms.

3.3.2.1 Mapping Matrix

Due to the fact that a graph can be described by an adjacency matrix, or in Carey's framework by a 1-step refinement matrix, the mapping between objects and ontology can be written in its matrix form too. That is, a matrix encoding the object-term mapping where the i-th row object maps to the j-th column term can be constructed. Formally,

Definition 3.24. *Let $\mathcal{T} = \{t_1, t_2, \ldots, t_n\}$ be a set of n terms in an ontology, and $\Omega = (\omega_1, \omega_2, \ldots, \omega_p)$ be a set of object identifiers. A **mapping matrix** is a matrix $\mathbf{M} = (m_{ij})_{p\times n}$ such that*

$$m_{ij} = \begin{cases} 1 & \textit{object } i \textit{ maps to term } j \\ 0 & \textit{otherwise.} \end{cases} \tag{3.14}$$

Now the main concept of Carey's philosophy can be defined.

3.3.2.2 Object-Ontology Complexes

Definition 3.25. *An* ***Object-Ontology Complex*** *is the ordered quadruple* $(\mathcal{T}, \Gamma, \Omega, \mathbf{M})$ *where* $\mathcal{T}$ *is a vocabulary,* Γ *is a refinement matrix encoding the ontology based on* $\mathcal{T}$*,* Ω *is a set of object identifiers, and* $\mathbf{M}$ *is a matrix mapping from* Ω *to* $\mathcal{T}$*.*

Thus, in the case of the Gene Ontology, the vocabulary $\mathcal{T}$ is one of the three ontologies (MF, BP or CC), Γ is the refinement matrix encoding the GO DAG, Ω is a query list of genes, and $\mathbf{M}$ is the matrix mapping genes to GO terms.
Example B.15 in appendix B shows the mapping matrix associated with the OOC B.7.

Using this notation and the truncated summation operation together, it is possible to define and compute many different characterizing methods and measures associated with ontologies. One of these characterizing methods is the *coverage matrix*.

3.3.2.3 Coverage Matrix

Definition 3.26. *Let* $(\mathcal{T}, \Gamma, \Omega, \mathbf{M})$ *be an Object-Ontology Complex. We say that a* ***term*** $t_j \in \mathcal{T}$ ***covers an object*** $\omega_i \in \Omega$ *if that term or any refinement of it is associated with the object via mapping* $\mathbf{M}$*.*

Note that for an object-term mapping matrix $\mathbf{M}$ and a refinement matrix Γ, the *boolean matrix product*[3] $\mathbf{C}_1 = \mathbf{M}\Gamma$ encodes the 1-step refinements of the mapping. That is, the (i, j)-element of $\mathbf{C}_1$ is 1 if term j is a 1-step refinement of the term to which object i is mapped by $\mathbf{M}$, and 0 otherwise.

Definition 3.27. *Let* $(\mathcal{T}, \Gamma, \Omega, \mathbf{M})$ *be an Object-Ontology Complex, and let* d_0 *be the depth of the ontology represented by* Γ*. We define the* ***coverage matrix*** *as*

$$\mathbf{C} = \bigoplus_{k=1}^{d_0+1} \mathbf{M}\Gamma^k = (c_{ij})_{p\times n} \tag{3.15}$$

[3] Given two matrices such that $A \in \mathcal{M}_{n\times p}(0,1)$ and $B \in \mathcal{M}_{p\times m}(0,1)$, we define the *boolean matrix product* is defined as

$$M \bigotimes N = \left(\bigvee_{k=1}^{n} (a_{ik} \wedge b_{kj}) \right) \in \mathcal{M}_{n\times m}(0,1).$$

See M. Castellet & I. Llerena ([25]).

where

$$c_{ij} = \begin{cases} 1 & \textit{object } i \textit{ is covered by term } j \\ 0 & \textit{otherwise.} \end{cases} \tag{3.16}$$

Example B.16 in appendix B shows the coverage matrix associated with the OOC B.7.

3.4 Main concepts of POSET theory

In previous sections a mathematical formalization was given based on the graph theory for the analysis of ontologies, and specially focused on the Directed Acyclic Graph of the GO introduced. A totally different approach adopted by Joslyn *et al.* ([85], [84], [83]) relies on the algebraic point of view of the order theory. This is a field of mathematics that is focused on different types of binary relations capturing the intuitive notion of a mathematical ordering ([135], [39], and [51]).

The main concept of Joslyn's approach is the *Partially Ordered Set* (POSET). This is a mathematical structure $\mathcal{P} = \langle P, \leq \rangle$ where P is a finite set and $\leq \subseteq P^2$ is a reflexive, antisymmetric, transitive binary relation P. Indeed POSETs are general combinatorial structures that are basically equivalent to Direct Acyclic Graphs. In spite of using different notations, both POSET theory and graph theory deal with the same objects. Figure 3.2 shows a classification of some related types of combinatorial objects, where POSETs, DAGs, or Trees are particular cases.

Directed graphs are much more specific structures than the trees. Every partially ordered set is a digraph without cycles. Semilattices, complete semilattices and trees are more specific posets. Notice that while nodes in lattices can have multiple parents, and of course in graphs with higher structural complexity, nodes in trees can not have multiple parents.

In the following subsections some definitions and concepts about lattices and order theory to deal with POSETs are introduced.

3.4.1 Partially Ordered Sets Definition

Definition 3.28. *A finite* ***partially ordered set****(or* ***POSET****) is a mathematical structure $\mathcal{P} = \langle P, \leq \rangle$ where P is a finite set and $\leq \subseteq P^2$ is a binary relation called* ***partial order*** *(or* ***order****) on P such that the relation is*

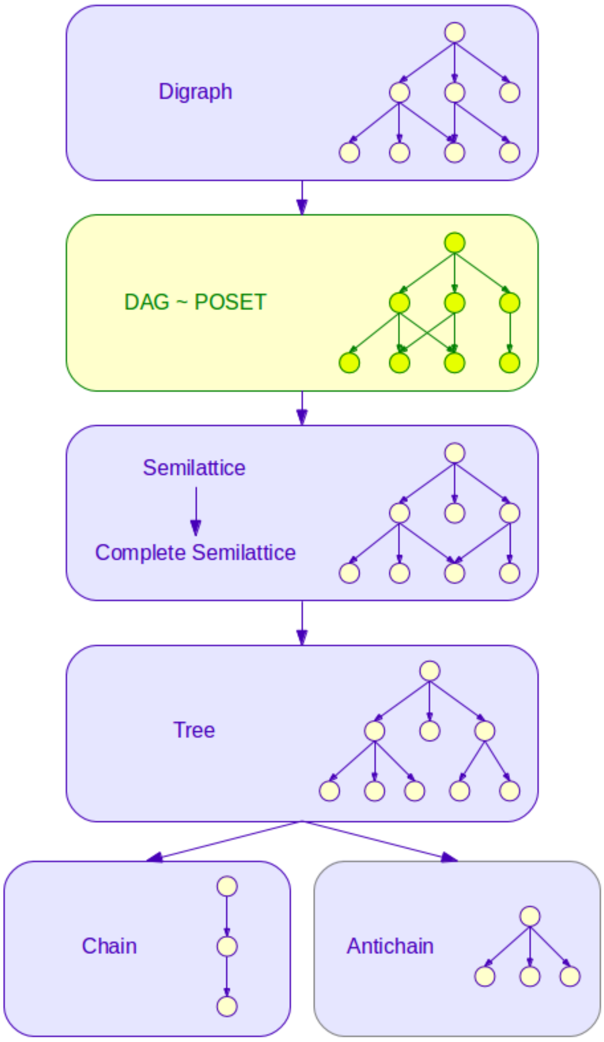


Figure 3.2: Classification of combinatorial objects.

1. *Reflexive:* $\forall p_i \in P,\ p_i \leq p_i$.
2. *Anti-symmetric:* $\forall p_i, p_j \in P,\ (p_i \leq p_j) \wedge (p_i \geq p_j) \Rightarrow p_i = p_j$.
3. *Transitive:* $\forall p_i, p_j, p_k \in P,\ (p_i \leq p_j) \wedge (p_j \leq p_k) \Rightarrow p_i \leq p_j$.

It is proved that every poset is a digraph with no cycles, and every tree or lattice is a poset. Moreover, each DAG determines a POSET based on the partial order on its nodes. $p_i \leq p_j$ exactly when there is a path from p_i to p_j in the DAG. However, notice that many different DAGs may give rise to this same reachability relation. For instance, a DAG with two edges e_{ik} and e_{kj} has the same reachability as a graph with three edges e_{ik}, e_{kj} and e_{ij}. Therefore, such a determination is not unique unless removing transitive nodes is considered.

3.4.2 Chains and Anti-chains

Definition 3.29. *Let $p_i, p_j \in P$ be two nodes in a poset. We say*

1. *p_i and p_j are **comparable** and we write $p_i \sim p_j \Leftrightarrow p_i \leq p_j$ or $p_i \geq p_j$.*
2. *A **chain** $C \subseteq P$ is a collection of comparable nodes.*
3. *The **height** $\mathcal{H}(\mathcal{P})$ is the size of the largest chain.*
4. *If C is a finite chain in P its **length** is the number $l(C) := |C| - 1$.*

The height of a poset is sometimes called *length*. “Opposite” concepts can be defined, similarly.

Definition 3.30. *Let $p_i, p_j \in P$ be two nodes in a poset. We say*

1. *p_i and p_j are **non-comparable** if $p_i \nsim p_j$.*
2. *An **anti-chain** is a collection of non-comparable nodes.*
3. *The **width** $\mathcal{W}(\mathcal{P})$ is the size of the largest anti-chain.*

3.4.3 Ideal, Filter and Hourglass

Definition 3.31. *Let $p_i \in P$ be any node in a poset. Then, we say that*

1. *The **ideal** (or **down-set**) of p_i is*

$$\downarrow p_i := \{p_j \in P : p_j \leq p_i\}. \tag{3.17}$$

2. *The* ***filter*** *(or* ***up-set****) of p_i is*

$$\uparrow p_i := \{p_j \in P : p_j \geq p_i\}. \tag{3.18}$$

3. *The* ***hourglass*** *of p_i is*

$$\Xi(p_i) :=\uparrow p_i \cup \downarrow p_i. \tag{3.19}$$

These concepts can be defined for a collection of nodes similarly as follows,

Definition 3.32. *Let $Q \subseteq P$ be a collection of nodes in a poset. We define*

$$\downarrow Q := \bigcup_{p_i \in Q} \downarrow p_i \quad , \quad \uparrow Q := \bigcup_{p_i \in Q} \uparrow p_i \quad , \quad \Xi(Q) := \bigcup_{p_i \in Q} \Xi(p_i). \tag{3.20}$$

Definition 3.33. *For any subset $Q \subseteq P$, let $p_i \in Q$ a node. Then, we say that p_i is a*

1. ***maximal*** *node in Q, if $\nexists\, p_j \in Q$ such that $p_j > p_i$, and*

2. ***minimal*** *node in Q, if $\nexists\, p_j \in Q$ such that $p_j < p_i$.*

And we define

1. *the* ***set of all maximal nodes in Q*** *as* $\max(Q)$, *and*

2. *the* ***set of all minimal nodes in Q*** *as* $\min(Q)$.

Notice that if Q is non-empty then both $\max(Q)$ and $\min(Q)$ are non-empty.

3.4.4 Upper and Lower Bounds

In some sense, for any given two nodes $p_i, p_j \in P$, the set $\uparrow p_i \cap \uparrow p_j$ is the *joint filter.*

Definition 3.34. *Let $p_i, p_j \in P$ be two nodes in a poset. Then,*

1. ***joins*** *of p_i and p_j are*

$$p_i \vee p_j := \min(\uparrow p_i \cap \uparrow p_j). \tag{3.21}$$

2. *The* ***meets*** *of p_i and p_j are*

$$p_i \wedge p_j := \max(\downarrow p_i \cup \downarrow p_j). \tag{3.22}$$

Again, these concepts can be defined for a collection of nodes.

Definition 3.35. *Given a collection of nodes $Q \subseteq P$ we define,*

$$\bigvee Q := \min\left(\bigcap_{p_i \in Q} \uparrow p_i\right) \quad , \quad \bigwedge Q := \max\left(\bigcup_{p_i \in Q} \downarrow p_i\right). \tag{3.23}$$

Definition 3.36. *Let $1 \in P$ a node such that $\max(P) = \bigvee(P) = \{1\}$, and $0 \in P$ a node such that $\min(P) = \bigwedge(P) = \{0\}$. Respectively, we say P is **upper-bounded** and **lower-bounded**.*

Definition 3.37. *Let $\mathcal{P} =< \mathcal{P}, \leq >$ a finite poset, and let $0, 1 \in P$ two nodes. The **closure** of $\mathcal{P}$ is*

$$\overline{\mathcal{P}} := \left\langle \mathcal{P} \cup \{0,1\}, \overline{\leq} \right\rangle, \tag{3.24}$$

where $\forall p_i, p_j \in P \;,\; p_i \overline{\leq} p_j \Leftrightarrow p_i \leq p_j$, and $\forall p_i \in P \;,\; 0 \overline{\leq} p_i \overline{\leq} 1$.

Most of the following notions have to be upper-, lower-, or completely bounded posets. From now on, when $\mathcal{P}$ is not bounded, we assume its closure $\overline{\mathcal{P}}$.

3.4.5 Interval Orders and Length

In discrete mathematics, when talking about intervals the first thought is for a set of points allocated on a piece of a "real line". However, notice that this "line" is not unique under the posets context. For instance, consider two comparable nodes $p_i \leq p_j$, here it might be the case that arbitrary chains of posets exist that join them.

Definition 3.38. *Let $\mathcal{P}$ be a finite poset. Given two comparable nodes $p_i \leq p_j$ in P an **interval** $[p_i, p_j]$ is defined as*

$$[p_i, p_j] := \{p_k \in P : p_i \leq p_k \leq p_j\} = \uparrow p_i \cap \downarrow p_j \equiv \mathcal{C}(p_i, p_j), \tag{3.25}$$

where $\mathcal{C}(p_i, p_j)$ is the set of all chains between p_i and p_j.

Definition 3.39. *Let $P_i, P_j \subseteq P$ be comparable subsets such that $\forall p_i \in P_i, p_j \in P_j \;,\; p_i \leq p_j$ (i.e. $P_i \leq P_j$). We define the interval $[P_i, P_j]$ as*

$$[P_i, P_j] := \bigcup_{\langle p_i, p_j \rangle \in P_i \times P_j} [P_i, P_j]. \tag{3.26}$$

Definition 3.40. *Let $\mathcal{C}(p_i, p_j)$ be the set of all chains between two comparable nodes $p_i \leq p_j$. Then*

1. *The* ***vector of chain lengths*** *is the collection of the lengths of all these chains*
$$h(p_i,p_j) := \langle |C(p_i,p_j)| \rangle. \tag{3.27}$$

2. *The* ***minimal chain length*** *is*
$$h_*(p_i,p_j) := \min_{C\in\mathcal{C}(p_i,p_j)} |C|. \tag{3.28}$$

3. *The* ***maximal chain length*** *is*
$$h^*(p_i,p_j) := \max_{C\in\mathcal{C}(p_i,p_j)} |C|. \tag{3.29}$$

3.4.6 POSET Ontology

In previous sections we talked about GO relationships. Specifically, in section 3.3.1, "is-a" and "part-of" links were introduced. The ontological structure of the GO could be thought of as a pair of DAGs: first by considering "is-a" links, and second by considering "part-of" links. Therefore, for each one of these relations the GO suggests a specific poset, either $\mathcal{P}_{is} =< P_{GO}, \leq_{is}>$ or $\mathcal{P}_{part} =< P_{GO}, \leq_{part}>$. But, in fact, these two kinds of links are considered to be equivalents. Thus, from the point of view of the POSET theory, a GO *poset model* arises defined as $\mathcal{P}_{GO} =< P_{GO}, \leq_{GO}>$ where $\leq_{GO} = \leq_{is} \cup \leq_{part}$.

Definition 3.41. *A* ***POSET Ontology*** *(or* ***POSO****) is a structure* $\mathcal{O} = \langle \mathcal{P}, X, F \rangle$*, where* $\mathcal{P} = \langle P, \leq \rangle$ *is a poset,* X *is a finite non-empty set of labels, and* F *is a function such that,*
$$\begin{array}{rccl} F: & X & \longrightarrow & 2^P \\ & x & \mapsto & F(x) \subseteq P, \end{array} \tag{3.30}$$
where 2^P *is the set of all functions from* P *to* $\{0,1\}$.[4]

Thus, in the GO, P is the collection of GO terms, $\leq$ is the ordering relations, and X is the set of gene products annotated in the GO terms.

Example B.17 in appendix B illustrates the idea of a POSO associated with the OOC B.7.

[4] Let P be a finite set with $|P| = n$ elements, and let us write any subset of P in the format $\{p_1, p_2, \ldots, p_n\}$ where p_i, $1 \leq i \leq n$, can take the value of 0 or 1. If $p_i = 1$, the i-th element of P is in the subset; otherwise, the $i-th$ element is not in the subset. Therefore, the number of distinct subsets of P is $|\mathcal{P}(P)| = 2^n$. This fact is the motivation for the notation 2^P and, note that there is a bijection between 2^P and the power set $\mathcal{P}(P)$. Hence 2^P and $\mathcal{P}(P)$ could be considered an identical set- theoretically ([70]).

3.5 Similarity Measures

A *similarity measure* is a function that quantifies the similarity between two objects. It is usually defined in some sense as the inverse[5] of a distance. There are many possible ways to build a similarity measure that depend on the nature of the objects compared and the goal being pursued. In bioinformatics, similarities are used in a wide diversity of applications. For instance, they are used for finding similar patterns in expression data ([28]) or for querying searches in gene sequence databases ([112]). In the context of biological ontologies, probably the most widely used measure of similarity for comparing functional annotation is the so-called *semantic similarity* ([72]).

The following subsections formally define the concepts of similarity measure and metric distance, and present the concept of semantic similarity measure, as well as an organization about different semantic similarity approaches.

3.5.1 Formal Definitions of Similarity Measure and Metric Distance

Definition 3.42. *Let $\Omega = \omega_1, \omega_2, \ldots, \omega_n$ be a set elements. A* ***similarity measure*** *between two elements $\omega_i, \omega_j \in \Omega$ is a function*

$$\begin{array}{rccl} s: & \Omega \times \Omega & \longrightarrow & [0,1] \in \mathbb{R} \\ & (\omega_i, \omega_j) & \mapsto & s(\omega_i, \omega_j) = s_{ij} \end{array} \tag{3.31}$$

such that

1. $0 \leq s_{ij} \leq 1$
2. $s_{ii} = 1$,
3. $s_{ij} = s_{ji}$.

$s(\omega_i, \omega_j)$ is a measure of the degree of similarity between the two elements ω_i and ω_j. That is, $s(\omega_i, \omega_j)$ increases as the similarity between ω_i and ω_j increases. So, the maximum similarity is reached when $s(\omega_i, \omega_j) = 1$.

The similarity concept is inversely related to distance concept. Thus, when working with similarity measures it is required to keep in mind that they are closely related with the concept of metric distances.

[5] By *inverse* we understand the idea of opposite (or reverse complementary), we are not referring to the mathematical concept of inverse function.

Definition 3.43. *Given a set of elements Ω, a **metric distance** is a function*

$$\begin{array}{rccl} d: & \Omega \times \Omega & \longrightarrow & \mathbb{R} \\ & (\omega_i, \omega_j) & \mapsto & d(\omega_i, \omega_j) = d_{ij} \end{array} \tag{3.32}$$

such that

1. $d(\omega_i, \omega_j) \geq 0 \ , \ \forall \omega_i, \omega_j \in \Omega$

2. $d(\omega_i, \omega_j) = 0 \Leftrightarrow \omega_i = \omega_j \ , \ \forall \omega_i, \omega_j \in \Omega$

3. $d(\omega_i, \omega_j) = d(\omega_j, \omega_i) \ , \ \forall \omega_i, \omega_j \in \Omega$

4. $d(\omega_i, \omega_j) \leq d(\omega_i, \omega_k) + d(\omega_j, \omega_k) \ , \ \forall \omega_i, \omega_j, \omega_k \in \Omega.$

Cuadras ([38]) suggested that $d_{ij} = 1 - sij$.

Therefore, in principle, a similarity should also obey the distance axioms, but in the opposite way. However, different theories suggest for violations in some or all of the axioms ([8]). In order to overcome these drawbacks different restrictions on the axioms have been proposed ([8]).

3.5.2 Semantic Similarity Measure

A semantic similarity measure can be considered a type of similarity measure, but with nuances and restrictions. The most significant restriction is that semantic similarities are intended for taxonomies (or hierarchies). The notion was suggested for measuring the strength of the semantic relationship between "units" of language, concepts or instances, through a numerical description obtained according to the comparison of information supporting their meaning or describing their nature ([72]). Harispe *et. al* ([72]) proposed the following definition:

> A *semantic similarity measure* quantifies the strength of the semantic relationships between two objects restricted to a taxonomical (or hierarchical) domain.

The general idea underlying semantic similarity measures is intuitive, *the more similar two concepts are, the greater their similarity, and the smaller the distance between the concepts.* But, more specifically, a semantic similarity measures the likeness of terms, concepts, words or any objects which can be characterized through semantics. That is, the likeness of compared objects is based on their meaning or semantic content, as opposed to similarity that can be computed with respect to the syntactical representation (i.e. its string formats). For instance, in transcriptomics, if the pathways associated with two lists of gene products are being compared, then the GO terms in which these genes were annotated could be considered, then a semantic similarity measure could be calculated to obtain a numerical value that would allow us to measure how their biological processes are related.

However, there is no unique way to formally define (i.e. in a mathematical sense) a semantic similarity. Depending on the field of study ([110]), the type of taxonomy ([72]) or the method used, ([128], [85], [122]) the definition of a semantic similarity measure can be different because of restrictions on the type of relationship which is conditioned by the context. In any case, mathematically speaking, most of the methods for computing semantic similarities consider the following definition:

Definition 3.44. *Given a set of elements Ω, a* ***semantic similarity*** *is a function*

$$\begin{array}{rccl} sim: & \Omega \times \Omega & \longrightarrow & [a,b] \subseteq \mathbb{R} \\ & (\omega_i, \omega_j) & \mapsto & sim(\omega_i, \omega_j) \end{array} \tag{3.33}$$

such that

1. $sim(\omega_i, \omega_j) \geq a \ , \ \forall \omega_i, \omega_j \in \Omega$
2. $sim(\omega_i, \omega_j) = b \Leftrightarrow \omega_i = \omega_j \ , \ \forall \omega_i, \omega_j \in \Omega$
3. $sim(\omega_i, \omega_j) = sim(\omega_j, \omega_i) \ , \ \forall \omega_i, \omega_j \in \Omega$
4. $sim(\omega_i, \omega_j) \leq sim(\omega_i, \omega_i), \ \forall \omega_i, \omega_j \in \Omega.$

The range of the codomain $[a, b] \subseteq \mathbb{R}$, usually, vary from 0 to 1, or from 0 to ∞, or from -1 to 1.

3.5.3 Organization and Classification of Semantic Similarities

There are many different ways to compute and classify semantic similarity measures ([72]). For example, they are usually organized according to the elements of the graph ([122]). This way to classify the semantic similarity measures distinguishes three approaches: (i) the edge-based approach, (ii) node-based approach and (iii) hybrid approach.

- **The edge-based approach** consists of measures focused on relationships between concepts. These measures try to find out how similar two concepts are by counting the number of edges that exist in the graph path between the corresponding nodes. For instance, a common procedure is to compute a distance by selecting either the shortest path or the average of all paths, when there is more than one path linking the two nodes, and then converting this distance into a similarity measure.

- **The node-based approach** consists of measures focused on the properties of the concepts being compared. These measures examine how similar the concepts are by taking into account properties that are attributable to the concepts themselves, their ancestors and/or their descendants. For example, one of the most widely used techniques is based on the Information Content (IC) ([128]). This measure calculates the amount of information that a concept conveys, allowing us to calculate how informative and specific such a concept is. IC can be computed in different ways. For instance, one of these methods is to count the number of offspring that a particular concept has. So, when two concepts in the ontology structure are being compared, the IC of common ancestors is used to give a measure of semantic similarity. To do this, there are two main approaches: (i) the *Most Informative Common Ancestor* (MICA) and (ii) the *Disjoint Common Ancestors* (DCA) ([122]).

- **The hybrid-based approach** considers measures that take advantage of both edge-based and node-based measures. That is, hybrid-based measures take into account different types of properties like the depth of the concept in the ontology, the number of children associated with the term, the IC, etc.

3.6 GO Terms and Semantic Similarity Measures

In section 3.5.3 it has been explained that there are several approaches to calculate semantic similarity measures ([15], [82], and [97]), and different classifications have been proposed to organize them ([72], and [122]). This section is devoted to introducing the main concepts of two of these strategies that are going to be used for discussing some of their properties later on, namely, the *node-based* approach and the *edge-based* approach.

Recapitulating on the main ideas of such methodologies, the *node-based approach* uses *Information Content* (IC) measures or information on object-part relationships to determine how similar two GO terms are, and the *edge-based approach* compares two terms by using the distance, or edge length, between the corresponding nodes. This part of the thesis is focused on two methods for computing semantic similarities, one from node-based approaches and the other from edge-based approaches. The first method was introduced by Phillip Lord *et al.* ([98]). From now on, it will be referred to as *Lord's measure*. Basically, they proposed a way to compute the similarity between two gene products based on the semantic similarity of their corresponding GO annotations, by considering a measure previously introduced by P. Resnik ([128]). Such a measure makes use of the IC of the terms. The second methods considered was developed by Joslyn *et al.* ([85]). From now on, it will be referred to as *Joslyn's measure*. It was proposed in a wider context of categorization problems of semantic hierarchies ([84]). One characteristic of this approach, which quickly begs comparison with Lord's measure, is that it rejects the idea of enriching a hierarchy, such as the GO, with IC. Instead it looks for alternative ways that take advantage of the structure of the hierarchy.

3.6.1 Lord's Measure

Lord *et al.* ([98]) introduced the idea of using the ontological annotations assigned to entries in biological databases in order to measure the similarities in terms of IC between these entries, which they called *semantic similarity*. That is, instead of attempting to define a similarity simply on the basis of the structure of the ontology, Lord *et al.* appealed to examine the usage of terms to find out how informative each term used is. Thus, instead of defining a new semantic similarity measure they adapted one that had been introduced by Resnik ([128]) to measure the similarity between terms in a semantic hierarchy based on the concept of Information Content.

3.6.1.1 The Information Content Concept

The coverage matrix defined in section 3.3.2.3 plays an important role for calculating the term informativeness and of course the semantic similarity. For instance, the sum $n(t)$ of column t from the coverage matrix is the number of times the term t or any of its refinements appears in the Object-Ontology Complex.

Definition 3.45. *Let* $(\mathcal{T}, \Gamma, \Omega, \mathbf{M})$ *be an Object-Ontology Complex, and let* $t \in \mathcal{T}$ *be any term in the ontology. The* ***probability of finding a term*** t

is a function such that

$$\begin{array}{rccl} P: & \mathcal{T} & \longrightarrow & [0,1] \\ & t & \mapsto & P(t) := \frac{n(t)}{N}, \end{array} \tag{3.34}$$

where $n(t)$ is the number of times that the term t or any of its specializations appears in the ontology and N is the total number of these terms that appear for the root node.

Note that the top node (i.e. the root term), t_0, verifies $P(t_0) = 1$.

As the probability increases, the informativeness of GO terms decreases. This is due to the fact that to be raising the level of abstraction of GO terms, these are increasingly referenced, and thus the informativeness of the terms is lower. That is, *the less often a term is used, the more informative it is.* Thus, if a GO term is rare and it has been "selected", then it gives much more information than a usually selected GO term. Reciprocally, if a selected GO term is often very uninformative, then it does not add very much to what we already know. To be more specific about the term "informative" the notion of information content is defined explicitly.

Definition 3.46. *The* ***information content*** *(*IC*) of a term t is a function measured as,*

$$\begin{array}{rccl} i: & \mathcal{T} & \longrightarrow & [0,\infty] \\ & t & \mapsto & i(t) := -\log P(t). \end{array} \tag{3.35}$$

In words, the IC of a single GO term is inversely proportional to its frequency in the GO DAG, and this frequency is propagated to the ancestors, meaning the IC of that GO term is related to its depth in the GO DAG. Thus, when a query list of genes is available, frequency is defined as the number of genes mapped to each GO term.

Example B.18 in appendix B shows the computation of the IC's associated with the OOC B.7.

3.6.1.2 Resnik's Similarity Measure

The IC of GO terms relies on the relationships given by the DAG structure of the ontology. Thus, the shared information between two terms is usually proportional to the IC of the *Most Informative Common Ancestor* (*MICA*) in the rooted DAG ([122]). That is, given GO terms t_i and t_j, the lowest common ancestor for both terms is the one with the greatest IC. Many different semantic similarity measures that rely on the MICA have been proposed

([72]). Resnik ([128]) introduced a measure of semantic similarity that is defined precisely in that way.

Definition 3.47. *The* ***semantic similarity of Resnik*** *between two terms $t_i, t_j \in \mathcal{T}$ in the ontology based on the IC, is defined as*

$$\begin{array}{rccl} sim_{Res}: & \mathcal{T} \times \mathcal{T} & \longrightarrow & [0, \infty] \\ & (t_i, t_j) & \mapsto & sim_{Res}(t_i, t_j) := \max\limits_{t \in S(t_i, t_j)} [i(t)], \end{array} \tag{3.36}$$

where $S(t_i, t_j)$ is the set of terms that subsumes both t_i and t_j, and $i(t)$ is the Information Content measure of term t.

Although the similarity is computed by using all upper bounds of the two terms, the information measured identifies the minimal upper bound. For instance, in the Object-Ontology Compex B.7 described in example B.14 (see appendix B), the set of terms that subsumes nodes H and E is $S(H, E) = \{I, B, C, 1\}$, but structurally the minimal upper bound is the MICA, I. MICA helps to differentiate cases where multiple inheritance is given. For instance, in the same figure the terms C and K are structurally indistinguishable as upper bounds of terms E and J, however the IC's can be completely different.

Example B.19 in appendix B shows the semantic similarities of Resnik between terms associated with the OOC B.7.

3.6.1.3 Lord's similarity measure

Resnik's measure selects only the one common ancestor. However, an ontology DAG allows multiple parents for each term, that is, two terms can share parents by multiple paths. Instead of using the one common ancestor between two GO terms t_i and t_j with the highest IC, Lord *et al.* introduced a variant of Resnik's measure that consists in taking the minimum $P(t)$ where there is more than one shared parent ([98]).

Definition 3.48. *The* ***semantic similarity of Lord*** *between two terms $t_i, t_j \in \mathcal{T}$ in the ontology is*

$$\begin{array}{rccl} sim_{Lord}: & \mathcal{T} \times \mathcal{T} & \longrightarrow & [0, \infty] \\ & (t_i, t_j) & \mapsto & sim_{Lord}(t_i, t_j) := sim_{Lord}(t_i, t_j) := -\log P_{ms}(t_i, t_j), \end{array} \tag{3.37}$$

where P_{ms} is the probability of the minimum subsumer

$$P_{ms}(t_i, t_j) := \min_{t \in S(t_i, t_j)} [P(t)] \tag{3.38}$$

where $S(t_i, t_j)$ is the set of terms that subsumes both t_i and t_j.

Resnik introduced a second measure where only the probability of a term is involved instead of the IC ([128]).

Definition 3.49. *The* ***semantic similarity of Resnik*** *between two terms* $t_i, t_j \in \mathcal{T}$ *in the ontology based on the probability, is defined as*

$$\begin{array}{rccl} sim_{P(t)}: & \mathcal{T} \times \mathcal{T} & \longrightarrow & [0,1] \\ & (t_i, t_j) & \mapsto & sim_{P(t)}(t_i, t_j) := \max\limits_{t \in S(t_i,t_j)} [1 - P(t)], \end{array} \tag{3.39}$$

where $S(t_i, t_j)$ *is the set of terms that subsumes both* t_i *and* t_j*, and* $P(t)$ *is the probability of finding the term* t*.*

3.6.2 Joslyn's Measure

Joslyn *et al.* argue that one drawback of Lord's measure is the fact that they rely on the probability of occurrence of terms, which not only depends on the ontology, but also on the dataset being considered, see ([85] and [84]). Thus, if the dataset changes, the probabilities will change too. These authors consider that this aspect is undesirable and they make an attempt to define appropriate measures of similarity which do not depend on any extra information beyond structural topology. Indeed, instead of similarities they use the complimentary concept of distances, or rather, in their case *pseudo-distance*, which is a function that does not completely verify all distance properties. Based on this concept, they present a method to score each term in an ontology. Thus, this ranking of nodes can be used as a way to summarize a list of terms, leading to a way to find the most characteristic term in an ontology.

In section 3.4.6 the concept of POSO was presented. Based on this structure, Joslyn *et al.* suggest different measures of distance that range from very intuitive measures (e.g. shortest paths ([127]), longest paths ([43]), and combinations of them ([72])) to others which are much more complex (such as an interval-valued distance ([163]), that take into account height and width of the structure). In order to discuss some aspects about this second strategy, which is an edge-based approach, this thesis is focused on the first types of measures. That is, measures based essentially on the length of paths between two nodes. These measures were introduced in the *GO Categorizer algorithms* ([85]). In short, given a list of GO terms associated with a concept, the GO categorizer algorithms return an ordered list of these GO terms according to their capability of representing overall semantic meanings of the set. It is performed by studying the relationship between the GO terms on the GO DAG through the POSET theory.

3.6.2.1 Pseudo-distances

The approach of Joslyn *et al.* begins with measures between comparable nodes $p_i \leq p_j$ (see section 3.4.2), which indicate how "high" p_j is above p_i.

Definition 3.50. *Let $h_*(p_i, p_j)$ and $h^*(p_i, p_j)$ be respectively the length of shortest and longest paths between p_i and p_j. A **pseudo-distance** is a function $\delta : P^2 \longrightarrow \mathbb{R}$ where*

$$h_*(p_i, p_j) \leq \delta(p_i, p_j) \leq h^*(p_i, p_j), \ \forall p_i \leq p_j \in P. \tag{3.40}$$

That is, a pseudo-distance is any function assigning a number to a given pair of comparable nodes in such a way that its value falls between the longest and the shortest path between these two nodes ([43]).

There is also a normalized form that measures what proportion of the height of the whole POSET $\mathcal{P}$ is taken up between nodes p_i and p_j.

Definition 3.51. *Let $(\mathcal{P})$ and $\mathcal{H}(\mathcal{P})$ a be POSET structure and the associated height. The **normalized pseudo-distance** is defined as*

$$\begin{array}{cccc} \overline{\delta}: & \mathcal{P} & \longrightarrow & [0,1] \\ & (p_i, p_j) & \mapsto & \overline{\delta}(p_i, p_j) := \frac{\delta(p_i,p_j)}{\mathcal{H}(\mathcal{P})}, \end{array} \tag{3.41}$$

The first pseudo-distances proposed by Joslyn *et al.* ([85]) were:

1. ***Minimum chain length***

$$\delta_m := h_*. \tag{3.42}$$

2. ***Maximum chain length***

$$\delta_x := h^*. \tag{3.43}$$

3. ***Average of extreme chain lengths***

$$\delta_{ax} := \frac{h_*(p_i, p_j) + h^*(p_i, p_j)}{2}. \tag{3.44}$$

4. ***Average of all chain lengths***

$$\delta_{ap} := \frac{\sum_{h \in \mathbf{h}(p_i, p_j)} h}{|\mathbf{h}|}. \tag{3.45}$$

Example B.20 in appendix B shows the pseudo-distances of the minimum chain length between terms associated with the OOC B.7.

Note that refinement matrix B.13 could have been used to compute the minimum chain length pseudo-distances between each pair of nodes.

3.7 `sims`: An `R` Package for Computing Semantic Similarities of an Ontology

`sims` is a package developed in `R` ([126]), and some specific functions from the packages `AnnotationDbi` ([119]), `expm` ([64]), `GOstats` ([54]), `plyr` ([161]), `Matrix` ([11]), `igraph` ([37]), `methods` ([126]), `plotrix` ([94]), `Rgraphviz` ([71]), and `vegan` ([118]).

Chapter 4

Results

In this chapter, the minor and main contributions of this part of the thesis are presented. These results are organized according to the sections outlined in *Material and Methods* 3.

4.1 Minor Contributions

The following subsections present two types of minor results: one, some parts of the proofs presented here were used as algorithms for programming some specific functions of the `R` package `sims`, and two, the mathematical formalization of some basic concepts that are usually relatively neglected in research and bioinformatic literature.

4.1.1 Graph Theory

The following propositions could be placed in section 4.2.4, because some pieces of their proofs have been used as parts of the algorithms in some functions of the package `sims`. However, due to the fact that they are results associated with graph theory, we found it convenient to present them in this section.

The first preliminary result could be skipped because it is merely an obvious formalism. However, we found it convenient to formalize the concept because it allowed us to take advantage of this idea when we were programming some specific functions during the development of the `R` package `sims`. Namely, in section 3.1.3, symmetric graphs were defined as those whose edges are connecting nodes. Then, it is easy to prove that the associated accessibility matrix is symmetric. Formally speaking:

Proposition 4.1. *If G is a symmetric graph $\Rightarrow$ $\mathbf{A}_G$ is symmetric.*

Proof. Let G be a symmetric graph. Then, all edges in G are target and source. Therefore, entries of its associated matrix are such that

$a_{ij} = a_{ji}$, $\forall i, j$. And vice versa, if $\mathbf{A}$ is symmetric, then $a_{ij} = a_{ji}$ is 1 or 0. Therefore, $e_{ij} = e_{ji}$, hence G is basically undirected, that is a symmetric graph. □

Therefore, in order to reduce the amount of information that some functions of the package `sims` pass to the memory of the computer for managing symmetric matrices, we only considered the lower triangular matrices rearranged as single vectors.

One of the most important characteristics of a DAG, or graphs in general, are the number of nodes and edges. These quantities allow computing of different types of semantic similarities. In node-based approaches, semantic similarities based on the IC are probably the most widely used. The IC is a measure that relies on the number of times that a term has been reached from an object (e.g. a gene product in the case of the GO). For this reason it is important to know how many links are reaching each term. Different methods to compute the number of edges that a node has have been proposed and studied ([162], [155], [17], [90]). Two of these methods have been studied. They are a well-known theorem of the graph theory, sometimes called the *Handshaking Theorem*, and corollary of it. According to the literature consulted, the proofs associated with them are either not reasoned from a matrix point of view ([155], [155]) or carelessly written ([17]). Therefore, we have strictly proven the theorem and the corollary based on the incidence matrix, in order to used the proofs for computing the number of edges in a DAG with the package `sims`.

Theorem 4.1 (Handshaking Theorem). *Let $G = (V, E)$ be a graph such that $|G| = n$ and $\varepsilon(G)$ the number of edges in G. Then,*

$$\sum_{i=1}^{n} d(v_i) = 2\varepsilon(G). \tag{4.1}$$

Proof. Let us consider the incidence matrix $\mathbf{B}_G$[1]. On one hand, the sum of the entries in each column is $2, \forall j = 1, 2, ..., \varepsilon(G)$, then

$$\sum_{j=1}^{\varepsilon(G)} 2 = 2\varepsilon(G).$$

[1] Note that, we are assuming G as a multigraph. In case of a DAG, we should consider the incidence matrix of the underlying graph for the DAG.

On other hand, the sum of entries in each row is $d(v_i),\ i = 1, 2, ..., n$. So

$$\sum_{i=1}^{n} d(v_i) \Rightarrow \sum_{i=1}^{n} d(v_i) = 2\varepsilon(G).$$

□

Corollary 4.1. *Let $D = (V, E)$ be a DAG such that $|D| = n$ and $\varepsilon(D)$ the number of edges in D. Then,*

$$\sum_{i=1}^{n} d^{+}(v_i) = \varepsilon(D) = \sum_{i=1}^{n} d^{-}(v_i). \tag{4.2}$$

Proof. Let us consider the incidence matrix $\mathbf{B}_D$. Handshaking Theorem says that the sum of entries by row is $d(v_i),\ i = 1, 2, ..., n$. Indeed, $d^{+}(v_i)$ is the sum of the positive one's and $-d_{-}(v_i)$ is the sum of negatives one's. Thus,

$$\sum_{i=1}^{n} d(v_i) = \sum_{i=1}^{n} d^{+}(v_i) - \sum_{i=1}^{n} d^{-}(v_i)$$

However, the sum of entries in each column is $0, \forall j = 1, 2, ..., \varepsilon(G) \Rightarrow \sum_{j=1}^{\varepsilon(G)} 0 = 0$. Then,

$$0 = \sum_{i=1}^{n} d^{+}(v_i) - \sum_{i=1}^{n} d^{-}(v_i) \Leftrightarrow \sum_{i=1}^{n} d^{+}(v_i) = \sum_{i=1}^{n} d^{-}(v_i)$$

Hence,

$$2\varepsilon = \sum_{i=1}^{n} d(v_i) = 2\sum_{i=1}^{n} d^{+}(v_i) \Leftrightarrow \varepsilon = \sum_{i=1}^{n} d^{+}(v_i).$$

□

4.1.2 Semantic Similarity Measures

While Joslyn *et al.* developed a well-defined mathematical framework [84] [85], and [83] based on order theory [39], and [135], in their approach, Lord *et al.* did not make a development that is so implicit, however, in their work. In order to minimize this lack of formalism, we suggest some basic formalisms in the following subsections.

4.1.2.1 The Information Content Concept

In previous sections, different methods for computing the semantic similarity between terms have been mentioned. The most intuitive idea relies on the length of the shortest path. That is, given multiple paths between two nodes that are being compared in an ontology, the length of the shortest path is the selected as the "best" measure of "similarity" [140]. Based on this measure, the closer two nodes are, the more similar they are. However, when dealing with ontology DAG structures, like the GO, this strategy is not enough to capture the essence of the annotations, as well as the relationships between them. In contrast to other graph structures, in ontology DAGs, the links between terms mean something. For instance, in the GO these links establish relations of either "is-a" or "part-of". In a broader sense, that is, if B is-a (or part-of) A, then B is a subset of A, $B \subseteq A$. Thus, taking into account this observation and the way of calculating the probability of finding a term t, the *monotonic property* of the probability theory can be proved *in terms of these relationships.*

Proposition 4.2 (Monotonic property)**.** *Let $t_i, t_j \in \mathcal{T}$ be two terms of refinement in an ontology. If t_i is-a (or part-of) $t_j \Rightarrow P(t_i) \leq P(t_j)$.*

Proof. Let $t_i, t_j \in \mathcal{T}$ such that $t_i \subseteq t_j$ (read t_i is-a t_j). Rewriting t_j as $t_j = t_i \uplus (t_j - t_i)$ where $\uplus$ is the disjoint union, then

$$P(t_j) = P(t_i \uplus (t_j - t_i)) = P(t_i) + \underbrace{P(t_j - t_i)}_{>0} \geq P(t_i)$$

□

In some sense, the paths between terms in an ontology act as channels that distribute the flow of information from more abstract to more specific terms.

Related with these probabilities, and as a node-based approach, Resnik introduced the Information Content concept ([128]). This measure captures the importance of the meaning of a single term, which is associated with the number of times that such a single term has been reached when a query object is available for an Object-Ontology Complex. Thus, it is very easy to prove that the root node of the Ontology is the term with the lowest IC because it is the most abstract concept.

Proposition 4.3. *$i(t_0) = 0$.*

Proof. Let $t_0 \in \mathcal{T}$ be the root term in the ontology. Then, based on the Monotonic property $P(t_0) = 1 \Rightarrow i(t_0) = 0$. □

4.2 Main Contributions

In the following subsections we present the main results of the thesis. First, we contribute with an important clarification about the relation between Lord's measure and Resnik's measure. Second, we suggest a proposition and a corollary for computing the number of times that the terms or any of their refinements appears in an OOC. Third, we proof that the Renik's measure in terms of distance and one of the Joslyn's pseudo-distances under certain conditions are actually metric distances.

4.2.1 GO Terms and Semantic Similarity Measures

Two different strategies for calculating semantic similarity measures have been considered. The first methodology relies on a node-based approach, which leads us to establish a relation between the list of objects (e.g. genes) and how terms of the ontology (e.g. GO terms) are related through an structure called Object-Ontology Complex (OOC). The second methodology relies on an edge-based approach. It is based on the POSET theory and suggests to construct a POSET Ontology (POSO) structure in order to establish the relation between objects and terms mentioned above. Thus, there exists a certain level of analogy between the OOC and the POSO. That is, concepts used in the definition of the OOC and in POSO can be easily related. Briefly, a POSET $\mathcal{P} = \langle P, \leq \rangle$ and the set of objectes (e.g. genes) X mapped by the function F can be described in terms of an OOC so that the vocabulary of the *Ontology* is the *POSET*, the *Object* is the set of *objects*, and the *function* $F(X)$ is the *mapping* between the Ontology and the list of objects. Thus, given a list of selected genes in an omic experiment, the OOC and the POSO can be considered two ways of formalizing mathematically “how to attribute biological meaning” when we are working with the GO. Note that this concept does not be confused with “biological significance” concept 1.1.

4.2.2 Lord's Measure

Clearly, as stated in section 3.6, an important thing for the similarity between two terms is the amount of information in common that they are sharing, which is indicated by a specific term that subsumes them. The concept of subsuming a set of refinements means explaining those terms in a more comprehensive concept. For instance, in figure B.7 of the example B.14 in appendix B nodes H and E are both subsumed by node I, whereas I is a more specific node that explains the information of nodes H and F. In this sense, Resnik ([128]) introduced a measure of similarity that depends on the MICA (see equation 3.36). Lord *et al.* argue that this semantic similarity measure selects only one common ancestor, and due to the fact that ontology DAGs (e.g. GO) allow multiple parents for each term, they suggested a variant of Resnik's measure that depends on the minimum $P(t)$ when there are more than one shared parents (see equation 3.37). However, in fact, this "new" measure of similarity is the same as the one proposed by Resnik ([128]) because

$$sim_{Res}(t_i,t_j) = \max_{t\in S(t_i,t_j)}[i(t)] = \max_{t\in S(t_i,t_j)}[-\log P(t)] \tag{4.3}$$

and since the logarithm is a monotone increasing function, we obtain that

$$= \ -\log \min_{t\in S(t_i,t_j)}[P(t)] = -\log P_{ms}(t_i,t_j) = sim_{Lord}(t_i,t_j).$$

Example B.22 in appendix B illustrates computationally the fact that Resnik' measure is the same than Lord's measure.

4.2.2.1 The Information Content Concept

In order to compute the IC, or rather, the number of times that a term t has been referenced, the product of the matrix with the number of paths of any length between each pair of terms by the mapping matrix can be used for computing the number of times that each term t or any of its specializations appears in the ontology. Therefore, we propose the following proposition and corollary.

Proposition 4.4. *Let* $(\mathcal{T},\Gamma,\Omega,\mathbf{M})$ *be an Object-Ontology Complex, where* $\mathcal{T}$ *is a set of* s *terms from a vocabulary,* Γ *is the refinement matrix encoding the ontology based on* $\mathcal{T}$*,* Ω *is a set of* p *object identifiers, and* $\mathbf{M}$ *is the matrix mapping from* Ω *to* $\mathcal{T}$*. Then, the matrix of the number of times that each term* t_i *or any of its specializations references to an specific ancestor* t_j *can be calculated as*

$$\mathbf{N}_t = \mathbf{M}(\mathbf{I}-\Gamma)^{-1}$$

Proof. On one hand, if $\mathbf{N}_t$ is the matrix where each of its elements n_{ij} is the number of times that an object identifier o_i in Ω maps to a term t_j and its specializations in $\mathcal{T}$, then it can be written as

$$n_{ij} = \sum_{k=0}^{r_j} n_{ij}^k = n_{ij}^0 + n_{ij}^1) + n_{ij}^2 + \ldots + n_{ij}^{r_j}$$

where n_{ij}^k denotes the number of times that an object identifier o_i maps to the kth- specialization of term t_j^k, and n_{ij}^0 is 1 if term t_j has been mapped by the object identifier o_i. On other hand, note that $(\mathbf{I}-\Gamma)(I+\Gamma+\Gamma^2+\ldots+\Gamma^r) = (\mathbf{I}-\Gamma^{r+1})$. The Neumann series ([145]) of a matrix holds that, when a matrix Γ has the property that

$$\lim_{r\to\infty} \Gamma^{r+1} = 0$$

then Γ is non-singular and its inverse may be expressed by the identity

$$(\mathbf{I}-\Gamma)^{-1} = \sum_{r=0}^{\infty} \Gamma^r$$

Thus, due to the fact that $\Gamma^{r+1} = 0$ when there are no paths with length exactly $k+1$, so the longest path is $< r+1$, the sum

$$\sum_{r=0}^{\infty} \Gamma^r = I + \Gamma + \Gamma^2 + \ldots + \Gamma^r$$

represents the number of paths of any length $\leq r$ between every pair of terms. Therefore,

$$\mathbf{M}(I + \Gamma + \Gamma^2 + \ldots + \Gamma^r) = \mathbf{M}(\mathbf{I}-\Gamma)^{-1}$$

is a matrix such that each of its elements is the number of times that an specific object reaches an specific term. That is, each of these elements is the number of times that each term t or any of its specializations references to an ancestor. Therefore,

$$\mathbf{N}_t = \mathbf{M}(\mathbf{I}-\Gamma)^{-1}.$$

□

Corollary 4.2. *Let $(\mathcal{T}, \Gamma, \Omega, \mathbf{M})$ be an OOC. Then, the number of times that a term t_j or any of its refinements appears in the OOC can be computed by summing the columns of the matrix N_t,*

$$n_j = \sum_{i=1}^{s} n_{ij}$$

where s is the number of terms in $\mathcal{T}$

Proof. Each element n_{ij} of matrix N_{ij} is the number of times that a term t_i reaches an ancestor t_j. Thus, $\forall j = 1, 2, \ldots, s$ where s is the number of terms in $\mathcal{T}$

$$n_j = n_{1j} + n_{2j} + \ldots + n_{sj}$$

is the number of times that each term t_1 or any of its specializations references to term t_j, plus the number of times that each term t_2 or any of its specializations references to term t_j, and so on until the number of times that each term t_s or any of its specializations references to term t_j. Hence, n_j is the number of times that all the term t_i $\forall i = 1, 2, \ldots, s$ or any of its specializations references to term t_j. □

Example B.21 in appendix B illustrates the application of the proposition and the corollary to the computation of the N_t matrix and $n(t)$ values.

4.2.2.2 Resnik's Measure in Terms of Metric Distance

As we mentioned in section 3.5.1, Cuadras ([38]) suggested that a metric distance can be computed in terms of a similarity $d_{ij} = 1 - sij$. Therefore, based on the second Resnik's measure 3.49, we prove the following proposition:

Proposition 4.5. *Given two terms t_i and t_j in $\mathcal{T}$ such that t_i is-a t_j then,*

$$d(t_i, t_j) = 1 - sim_{P(t)}(t_i, t_j) \tag{4.4}$$

is a metric distance.

Proof. Let $t_i, t_j \in \mathcal{T}$ be two terms in the ontology such that $t_i \subseteq t_j$. Then,

$$\begin{aligned} d(t_i, t_j) &= 1 - sim_{P(t)}(t_i, t_j) = 1 - \left\{ \max_{t \in S(t_i, t_j)} [1 - P(t)] \right\} \\ &= 1 - \left\{ 1 - \min_{t \in S(t_i, t_j)} P(t) \right\} \\ &= \min_{t \in S(t_i, t_j)} P(t) \end{aligned}$$

Therefore, if this function $d : \mathcal{T} \times \mathcal{T} \longrightarrow [0, 1] \subseteq \mathbb{R}$ complies with the axioms of metric distance, we will have certainly proved that it is a metric distance. So,

1. $d(t_i, t_j) \overset{?}{\geq} 0 \ , \ \forall t_i, t_j \in \mathcal{T}$

$$0 \leq P(t) \leq 1 \ \Rightarrow \ d(t_i, t_j) = \min_{t \in S(t_i, t_j)} P(t) \geq 0.$$

2. $d(t_i, t_j) = 0 \overset{?}{\Leftrightarrow} t_i = t_j \ , \ \forall t_i \in \mathcal{T}$

   $\Leftarrow)$ $S(t_i, t_i) = \emptyset \Rightarrow d(t_i, t_i) = \min_{t \in S(t_i, t_i)} P(t) = 0.$

   $\Rightarrow)$ $0 = d(t_i, t_j) = \min_{t \in S(t_i, t_j)} P(t) \Rightarrow P(t) = 0 \Rightarrow n(t) = 0$

   $\Rightarrow S(t_i, t_j) = \emptyset \Leftrightarrow t_i = t_j.$

3. $d(t_i, t_j) \overset{?}{=} d(t_j, t_i) \ , \ \forall t_i, t_j \in \mathcal{T}$

   $S(t_i, t_j) = \{t \in \mathcal{T} : \textit{subsuming } t_i \textit{ and } t_j\} = S(t_j, t_i)$ then,

$$d(t_i, t_j) = \min_{t \in S(t_i, t_j)} P(t) \quad = \quad \min_{t \in S(t_j, t_i)} P(t) = d(t_j, t_i)$$

4. $d(t_i, t_j) \overset{?}{\leq} d(t_i, t_k) + d(t_j, t_k) \ , \ \forall t_i, t_j, t_k \in \mathcal{T}$

   On the left hand of the inequality we have that,

$$d(t_i, t_j) \quad = \quad \min_{t \in S(t_i, t_j)} P(t),$$

   and on the right hand,

$$d(t_i, t_k) + d(t_j, t_k) \quad = \quad \min_{t \in S(t_i, t_k)} P(t) + \min_{t \in S(t_j, t_k)} P(t)$$

   Thus, we need to prove that,

$$\min_{t \in S(t_i, t_j)} P(t) \quad \overset{?}{\leq} \quad \min_{t \in S(t_i, t_k)} P(t) + \min_{t \in S(t_j, t_k)} P(t)$$

   Then, we have to distinguish two cases:

Case 1: $t_i \subseteq t_j \subseteq t_k$ (we would proceed similarly for $t_i \subseteq t_j \subseteq t_k$) or $t_i \subseteq t_j \subseteq t_l$ such that $t_k \subseteq t_l$

$$S(t_i, t_k) \subseteq S(t_i, t_j) \quad \Rightarrow \quad \min_{t \in S(t_i, t_j)} P(t) \leq \min_{t \in S(t_i, t_k)} P(t)$$

Case 2: $t_i \subseteq t_k \subseteq t_j$ or $t_i \subseteq t_l \subseteq t_j$ such that $t_k \subseteq t_l$

$$S(t_i, t_j) = S(t_j, t_k) \quad \Rightarrow \quad \min_{t \in S(t_i, t_j)} P(t) = \min_{t \in S(t_j, t_k)} P(t)$$

Therefore, $\forall t_i, t_j, t_k \in \mathcal{T}$

$$\min_{t \in S(t_i,t_j)} P(t) \quad \leq \quad \min_{t \in S(t_i,t_k)} P(t) + \min_{t \in S(t_j,t_k)} P(t).$$

□

Note that, if the monotonic property is not satisfied, then it is not possible to guaranteeing the triangle inequality.

4.2.3 Joslyn's Measure

Pseudo-distances are useful measures for calculating how "different" two in a POSET are. However, these types of measures cannot always be used. Pseudo-distances make sense only when measuring *comparable* terms, that is, between two terms where one is a specialization (i.e. a refinement in terms of Carey's vocabulary) of the other. Therefore, pseudo-distances cannot measure *non-comparable* terms, and this is the major difference with Lord's measure. However, when restricting to comparable terms, a pseudo-distance is actually a distance. Here, we present a proposition and proof of this hypothesis, focused on the pseudo-distance of the minimum chain length 3.42.

Proposition 4.6. *Let $p_i, p_j \in P$ be two comparable nodes such that $p_i \leq p_j$. The pseudo-distance*

$$\begin{array}{rrcl} \delta_m : & P \times P & \longrightarrow & \mathbb{N} \cup \{0\} \subseteq \mathbb{R} \\ & (p_i, p_j) & \mapsto & \delta_m(p_i, p_j) := \min\limits_{C \in \mathcal{C}(p_i,p_j)} |C| = h_*(p_i, p_j) \end{array} \tag{4.5}$$

where $\mathcal{C}(p_i, p_j)$ is the set of all chains between two nodes p_i and p_j, is a metric distance.

Proof.

1. $\delta_m(p_i, p_j) \overset{?}{\geq} 0 \ , \ \forall p_i \sim p_j \in P$

Case 1: $\mathcal{C}(p_i, p_j) \neq \emptyset \Rightarrow |C| \in \mathbb{N} \ , \ \forall C \in \mathcal{C}(p_i, p_j)$

$$\Rightarrow \delta_m(p_i, p_j) = \min_{C \in \mathcal{C}(p_i,p_j)} |C| > 0.$$

Case 2: $\mathcal{C}(p_i, p_j) = \emptyset \Rightarrow |C| = 0 \ , \ \forall C \in \mathcal{C}(p_i, p_i)$

$$\Rightarrow \delta_m(p_i, p_j) = \min_{C \in \mathcal{C}(p_i,p_j)} |C| = 0.$$

2. $\delta_m(p_i,p_j)=0 \overset{?}{\Leftrightarrow} p_i=p_j \ , \ \forall p_i \sim p_j \in P$

$$\begin{aligned} 0 &= \delta_m(p_i,p_j)=\min_{C\in\mathcal{C}(p_i,p_j)}|C| \Leftrightarrow |C|=0, \ \forall C\in\mathcal{C}(p_i,p_j) \\ &\Leftrightarrow \mathcal{C}(p_i,p_j)=\emptyset \Leftrightarrow p_i=p_j \ , \ \forall p_i\sim p_j\in P. \end{aligned}$$

3. $\delta_m(p_i,p_j) \overset{?}{=} \delta_m(p_j,p_i) \ , \ \forall p_i \sim p_j \in P$

$\mathcal{C}(p_i,p_j)=\{\textit{chains between } p_i \textit{ and } p_j\}=\mathcal{C}(p_j,p_i)$ then,

$$\delta_m(p_i,p_j)=\min_{C\in\mathcal{C}(p_i,p_j)}|C| \ = \ \min_{C\in\mathcal{C}(p_j,p_i)}|C|=\delta_m(p_j,p_i) \ , \ \forall p_i\sim p_j\in P.$$

4. $\delta_m(p_i,p_j) \overset{?}{\leq} \delta_m(p_i,p_k)+\delta_m(p_j,p_k) \ , \ \forall p_i,p_j,p_k \in P$ comparable

On the one hand we have that,

$$\delta_m(p_i,p_j) \ = \ \min_{C\in\mathcal{C}(p_i,p_j)}|C| \ ,$$

and on the other,

$$\delta_m(p_i,p_k)+\delta_m(p_j,p_k) \ = \ \min_{C\in\mathcal{C}(p_i,p_k)}|C|+\min_{C\in\mathcal{C}(p_j,p_k)}|C|$$

Then, we have to prove that,

$$\min_{C\in\mathcal{C}(p_i,p_j)}|C| \ \overset{?}{\leq} \ \min_{C\in\mathcal{C}(p_i,p_k)}|C|+\min_{C\in\mathcal{C}(p_j,p_k)}|C|.$$

Now, we know that, $p_i\sim p_j \Leftrightarrow p_i\leq p_j$ or $p_i\geq p_j$. Thus, we distinguish three cases

Case 1: $p_i\leq p_j\leq p_k$ (or $p_j\leq p_i\leq p_k$)

$$\min_{C\in\mathcal{C}(p_i,p_k)}|C|\geq\min_{C\in\mathcal{C}(p_i,p_j)}|C|\Rightarrow\min_{C\in\mathcal{C}(p_i,p_k)}|C|+\min_{C\in\mathcal{C}(p_j,p_k)}|C|\geq\min_{C\in\mathcal{C}(p_i,p_j)}|C|.$$

Case 2: $p_k\leq p_i\leq p_j$ (or $p_k\leq p_j\leq p_i$)

We will proceed similarly to *Case 1.*

Case 3: $p_i \leq p_k \leq p_j$ (or $p_j \leq p_k \leq p_i$)

Let $\mathcal{C}^{p_k}(p_i, p_j)$ be the set of chains between p_i and p_j through the p_k node. Then, by construction of this set,

$$\min_{C \in \mathcal{C}(p_i,p_k)} |C| + \min_{C \in \mathcal{C}(p_j,p_k)} |C| \geq \min_{C \in \mathcal{C}^{p_k}(p_i,p_j)} |C| \geq \min_{C \in \mathcal{C}(p_i,p_j)} |C|.$$

Therefore, $\forall p_i, p_j, p_k \in P$ comparable

$$\min_{C \in \mathcal{C}(p_i,p_j)} |C| \quad \leq \quad \min_{C \in \mathcal{C}(p_i,p_k)} |C| + \min_{C \in \mathcal{C}(p_j,p_k)} |C|.$$

□

Similarly, we could prove that, assuming comparable terms, the other distances proposed are metric distances.

4.2.4 `sims`: An `R` Package for Computing Semantic Similarities of an Ontology

At the beginning of this part of the thesis (see section II) an ontology was introduced as a way for annotating concepts of a certain domain, and the vocabulary of an ontology is arranged as a rooted DAG 3.13.

Very often measuring the relationship between pairs of terms and ontology is required ([124], [127]). In previous sections we have seen that an appropriate measure for assessing these relationships relies on the semantic similarity measures ([60]). We have also seen that there are many different methods and approaches for computing semantic similarities ([72]).

In order to compute semantic similarity measures an `R` ([126]) package called `sims` (for `s`emantic `si`milarity `m`easure`s`) has been developed.

This section is addressed to present the package `sims`. Examples of the main possibilities of `sims` are shown in the vignette of the package, which is provided in appendix A.

4.2.4.1 Availability of `sims`

The package is freely available under a License GPL-2 (`http://www.r-project.org/Licenses/GPL-2`). It can be downloaded from the GitHub repository `https://github.com/jlmosquera/sims`.

4.2.4.2 Requirements

To use the package one must have `R 3.1.1` (or greater) ([126]) installed, as well as `Bioconductor 2.14` ([63]) (or greater). But also, some extra packages from CRAN (`http://CRAN.R-project.org/`) ([126]) are required. Specifically, `expm` ([64]), `plyr` ([161]), `Matrix` ([11]), `igraph` ([37]), `plotrix` ([94]), and `vegan` ([118]).

4.2.4.3 Main Possibilities of the Package and the List of Functions

Main possibilities of this package are:

1. To deal with ontology structures (i.e. Object-Ontology Complex).

2. To compute semantic similarities between terms of an ontology.

3. To compute semantic similarity profiles between GO terms.

4. To compare semantic similarity profiles between GO terms associated with two lists of Gene Entrez Identifiers.

These tasks can be performed thanks to a list of 51 functions that has been implementes in the package. These functions are shown in table 4.1, where they are organized by groups of possibilities:

4.2.4.4 Semantic Similarity Measures Implemented in `sims`

`sims` package consists of fourteen methods from the node-based and edge-based approaches 3.5.3. Specifically, there are implemented seven semantic similarity measures from node-based approaches proposed by Resnik ([128]), Lin ([97]), Schlicker *et al.* ([134]), Jiang and Conrath ([82]), Mazandu and Mulder ([107]), Pirró and Seco ([124]), and Pirró and Euzenat ([123]) are implemented in `sims` (see table 4.2). Most of these measures are based on the IC 3.6.1.1 and MICA 3.6.1.2. With regard to edge-based approaches there are implemented two semantic similarity measures proposed by Resnik ([128]), and Rada *et al.* ([127]), one distance measure proposed by Rada ([127]) and four pseudo-distances proposed by Joslyn *et al.* ([84], [85]).

Table 4.2 shows the list of the measures implemented in the package and some basic descriptions about them.

sims	General description of the package {sims}
Functions for managing an OOC	
ancestors	Ancestors for each term of the ontology
commonAncestors	Common ancestors for each pair of terms of the ontology
depth	Depth of the ontology
getA	Accessibility matrix associated with the DAG structure of the ontology
getGk	Builds a list of the matrices with the number of paths between each pair of terms that are directly connected for each length
getGr	Number of paths of any length between each pair of terms that are directly connected
inverseIminusG	Computes the number of paths of any length between each pair of terms in the ontology
is.OOC	Tests if its argument is a (strict) OOC object
Nt	Number of times that each term or any of its specializations references to an ancestor
simsMat	Coerces a 1-column 'data.frame' resulting from semantic similarity functions to be an object of class 'dist'.
OOC	General container for an Object-Ontology Complex (OOC)
termPairs	Builds the pairs of different terms or characters
toMat	Builds a matrix of zero and one elements such that zero indicates there is no a relation between row and column, and one there is a relation.
toOOC	Builds an Object-Ontology Complex (OOC)
toPairs	Builds a 2-columns 'data.frame' relating elements of a matrix with value one.
Functions for computing semantic similarities between terms of an ontology	
simFaith	Semantic similarity of Pirro and Euzenat for each pair of terms
simJC	Semantic similarity of Jiang and Conrath for each pair of terms
simLin	Semantic similarity of Lin for each pair of terms
simNunivers	Semantic similarity of Mazandu and Mulder for each pair of terms
simPsec	Semantic similarity of Pirro and Seco for each pair of terms
simRada	Semantic similarity measure of Rada _et al._ for each pair of terms
simRel	Semantic similarity of Schlicker _et al._ for each pair of terms
simRes	Semantic similarity of Resnik for each pair of terms
simRes.eb	Semantic similarity measure of Resnik _et al._ for each pair of terms, considering the maximal depth of the ontology
distRada	Distances of the shortest paths between each pair of terms in the ontology
pdHap	Pseudo-distances of the average of all chain lengths between comparable terms of the ontology.
pdHax	Pseudo-distances of the average of extreme chain lengths between comparable terms of the ontology.
pdHm	Pseudo-distance of the minimum chain lengths between comparable terms of the ontology.
pdHx	Pseudo-distance of the maximum chain lengths between comparable terms of the ontology.
sims.eb	Wrapper function that calls different methods for computing semantic similarities based on edge-based approaches
sims.nb	Wrapper function that calls different methods for computing semantic similarities based on node-based approaches
pseudoDists	Wrapper function that calls different methods for computing pseudo-distances
resnikSummary	Summary table providing with the number of times that each term or any of its refinement appears in the OOC, the probability of finding the term, and the Information Content of the term
ICA	Information Content (IC) of common ancestors
LCAs	Length of the shortest paths containing the Least Common Ancestors (LCA) between each pair of terms
summaryMICA	Computes for each pair of terms the Information Content (IC) of each term the Most Informative Common Ancestor (MICA), and the subsumer associated with the MICA
summaryPaths	Lengths of the chains (in terms of depth) or number of paths between each pair of terms.
summarySims	Summary of semantic similarity estimates between each pair of terms and measure
Functions for computing semantic similarities profiles of GO terms	
goOOC	Builds an Object-Ontology Complex (OOC) whose slots are associated with GO Identifiers
gosims	Wrapper function that calls different approaches and methods for computing semantic similarities between GO Identifiers given a list of Entrez Gene IDs
gosimsAvsB	Wrapper function for computing semantic similarities between GO Identifiers for two lists of Entrez Gene IDs
mapEG2GO	Mapping Entrez Gene IDs to Gene Ontology IDs
mappingMatrix	Mapping matrix from the Entrez Gene IDs to the GO IDs associated with the directed subgraph extracted from GO DAG structure
refinementMatrix	Builds the refinement matrix associated with the DAG structure of Gene Ontology
simsBetweenGOIDs	Wrapper function that calls different approaches and methods for computing semantic similarities between GO ID ancestors of a list of GO ID's
Functions for comparing semantic similarities profiles of GO terms associated with two lists of Gene Entrez Ids.	
cosSim	Cosine similarity measure
gosimsProfiles	Plots a vertical bar diagram whose bars are associated with the semantic similarities between each pair of terms, and such that bars on the left side of the plot are the corresponding to the first group of objects and on the bars on the right side are the bars corresponding to the second group of objects
plotGODAG	Plots a subgraph from the GO associated with one or two lists of Entrez Gene Identifiers
plotHistSims	Histogram of two semantic similarity profiles
summarySimsAvsB	Summary of a two-columns matrix with semantic similarity estimates between each pair of terms for the same measure

Table 4.1: List of functions in sims package.

Author	Approach	Type	Codomain	Formula	Comment	Reference
Jiang and Conrath	NB	sim	$[0,1]$	$sim_{JC}(t_i,t_j) = \frac{1}{1+IC(t_i)+IC(t_j)-(2MICA)}$	Based on the taxonomic distance computed as a function of the IC	[82]
Lin	NB	sim	$[0,1]$	$sim_{Lin}(t_i,t_j) = \frac{2IC(MICA)}{IC(t_i)+IC(t_j)}$	Based on the IC of the MICA of the compared terms divided by the sum of the Ics of the compared terms	[97]
Mazandu and Mulder	NB	sim	$[0,1]$	$sim_{Nunivers}(t_i,t_j) = \frac{IC(MICA)}{\max IC(t_i),IC(t_j)}$	Based on the IC of MICA of the compared terms divided by the maximal IC of the compared terms	[107]
Pirró and Euzenat	NB	sim	$[0,1]$	$sim_{Faith}(t_i,t_j) = \frac{IC(MICA)}{IC(t_i)+IC(t_j)-IC(MICA)}$	Based on the Jaccard coefficient taking into account the IC of the MICA of the compared therms and the Ics of the compared terms	[123]
Pirró and Seco	NB	sim	$[-m,2m]$	$sim_{Psec}(t_i,t_j) = 3IC(MICA) - IC(t_i) - IC(t_j)$	m is the maximal term IC	[124]
Resnik	NB	sim	$[0,\infty]$	$sim_{Res}(t_i,t_j) = IC(MICA) = \max_{t\in S(t_i,t_j)}(IC(t))$	Based on the MICA. The range depends on the IC	[128]
Schlicker *et al.*	NB	sim	$[0,1]$	$sim_{Rel}(t_i,t_j) = sim_{Lin}(t_i,t_j)(1-P(MICA))$	Based on the Lin measure taking specificity into account	[134]
Rada distance	EB	dist	$[0,\infty]$	$dist_{Rada}(t_i,t_j) = sp(t_i,t_j,LCA)$	Based on the shortest path constrained to contain the LCA of the compared terms.	[127]
Rada	EB	sim	$[0,\infty]$	$sim_{Rada}(t_i,t_j) = \frac{1}{1+dist_{Rada}(t_i,t_j)}$	Base on the Rada distance	[127]
Resnik	EB	sim	$[0,2d]$	$sim_{Res-ed}(t_i,t_j) = 2*(\max depth) - (dist_{Rada}(t_i,t_{LCA}) + dist_{Rada}(t_j,t_{LCA}))$	Based on the Rada distance which has been bounded by (twice) the max depth of the ontology (i.e. d)	[128]
Joslyn *et al.* (δ_m)	EB	pdist	$[0,\mathcal{H}(P)]$	$\delta_m(t_i,t_j) = h_*(t_i,t_j) = \min_{C\in\mathcal{C}(t_i,t_j)}\|C\|$	Based on the minimum chain length. $\mathcal{H}(P)$ is the height of the poset	[84], [85]
Joslyn *et al.* (δ_x)	EB	pdist	$[0,\mathcal{H}(P)]$	$\delta\_x(t_i,t_j) = h^*(t_i,t_j) = \max_{C\in\mathcal{C}(t_i,t_j)}\|C\|$	Based on the maximum chain length. $\mathcal{H}(P)$ is the height of the poset	[84], [85]
Joslyn *et al.* (δ_{ax})	EB	pdist	$[0,\mathcal{H}(P)]$	$\delta_{ax}(t_i,t_j) = \frac{h_*(t_i,t_j)+h^*(t_i,t_j)}{2}$	Based the average of extreme chain lengths. $\mathcal{H}(P)$ is the height of the poset	[84], [85]
Joslyn *et al.* (δ_{ap})	EB	pdist	$[0,\mathcal{H}(P)]$	$\delta_{ap}(t_i,t_j) = \frac{\sum_{h\in\mathbf{h}(t_i,t_j)} h}{\|\mathbf{h}\|}$	Based on the average of all chain lengths. $\mathcal{H}(P)$ is the height of the poset	[84], [85]

NB: Node-Based approach, **EB**: Edge-Based approach. **sim**: semantic similarity; **dist**: distance; **pdist**: pseudo-distance

Table 4.2: Semantic similarity measures, distances and pseudo-distances implemented in the package `sims`.

4.2.5 Semantic Similarity Profiles between GO Terms

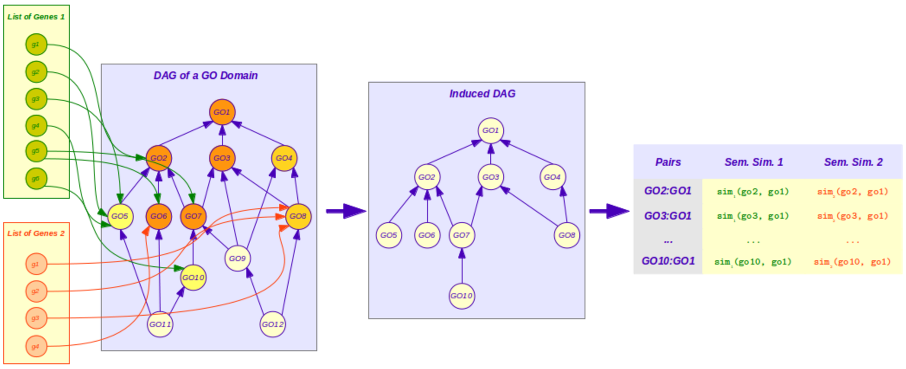
As mentioned above 3.1.5, one of the possibilities implemented in `sims` package is addressed to compare lists of Gene Entrez Identifiers annotated in the GO.

Sometimes a researcher is faced with the problem of comparing two lists of genes. One of the possibilities for performing this task is to considering the biological annotation of the GO, and then calculate the so-called *functional similarity measures* ([134], [157], [48], [122], [72]). That is, given two lists of genes, he/she looks for the functional annotations associated with each list of genes respectively and, then, the researcher measures how these two lists of genes are similar by comparing the similarity between the two lists of functional annotations. In other words, this approach is addressed to know how well a measure captures the similarity in function between these lists of genes. However, this question is not trivial, because there is no fashion solution for determining the true functional similarity between two lists of genes. Different measures have been proposed, but they are still a subject of debate ([122]). Even so, different tools have implemented such measures. In this regard, instead of providing an specific measure, we have considered an alternative approach based on what we called semantic similarity profile.

Given an ontology, we define a *semantic similarity profile* as the list of semantic similarity measures between all the pairs of terms from an induced subgraph given by a list of selected objects. Thus, when we focused on the GO and we would like to compare two lists of genes, the idea is to map these lists of genes to the GO, and compute the semantic similarities between all the pairs of GO terms based on this common induced subgraph but twice, one, the semantic similarities based on the first lists of genes and, two, based on the second list of genes. Therefore, we obtain two different lists of semantic similarity profiles. Figure 4.1 shows the schema for two hypothetical lists of genes and a fake of the GO.

Then, with these semantic similarity profiles in hand, we have implemented some functions for yielding a summary that consists of:

1. An statistic descriptive for each profile of semantic similarity measures.
2. A Mantel's Test ([101]) for examining the association between the distance matrices (i.e. the similarity matrices).
3. The Cosine Similarity ([147]) for determining the similarity between



Pairs	Sem. Sim. 1	Sem. Sim. 2
GO2:GO1	$sim_1(go2, go1)$	$sim_2(go2, go1)$
GO3:GO1	$sim_1(go3, go1)$	$sim_2(go3, go1)$
...	...	...
GO10:GO1	$sim_1(go10, go1)$	$sim_2(go10, go1)$

Figure 4.1: Schema for comparing two semantic similarities profiles associated with the two lists of genes respectively.

the two semantic similarity profiles.

This summary is also accompanied by three plots:

1. An histogram of the semantic similarity profiles that shows both distribution in the same figure.

2. A vertical bar diagram, whose bars are associated with the semantic similarities between each pair of terms.

3. An induced subgraph of the GO domain associated with (one or) both lists of Entrez Gene Identifiers.

Chapter 5

Discussion

This part of the thesis focused on the exploration of two different semantic similarity approaches to deal with Gene Ontology (GO) ([148], [149], and [150]) terms, in order to give a biological interpretation to the resulting findings associated with high-throughput data generated in omic experiments. The research tried to show that both approaches are related to the concept of metric distance, on the one hand, and, on the other hand, to developing an `R` package for computing semantic similarity measures between ontology terms and comparing semantic similarity profiles.

There is no unique methodology for giving biological meaning to a given list of genes. We saw that depending on the approach used to synthesize the mapping between the genes list and the Gene Ontology (GO), graph theory ([43], [17]) may be helpful for answering common questions (see section 3.2.2). But there are also other alternatives like the Partially Ordered Sets (POSET) theory ([135], [39], and [51]). The first approach that we studied was framed in graph theory and was a semantic similarity measure proposed by Lord et al. ([98]). It is one of the so called node-based approaches (see section 3.5.3). The second measure was a pseudo-distance, which is framed in POSET theory, and was proposed by Joslyn et al. ([85], [84], [83]). This measure is an edge-based approach (see section 3.5.3). So, in order to deal with a complex structure such as the GO, large lists of concepts about both theories were introduced.

When we focused our effort in the Lord's measure we realized that a step backwards was necessary. After surveying graph theory concepts, our research led us to focus on Carey's framework ([23]). Carey took advantage of the definition of an ontology and described relationships between terms as refinements. But, we emphasized that refinements are an alternative way to describe the relationships between nodes in a graph (see section 3.3.1). Likewise, refinement matrix and accessibility matrix concepts are both of interest because they are associated with the graph structure of an ontology and many relevant measures may be derived from them. Based on these concepts, Carey suggested the idea of an Object-Ontology Complex (OOC). This structure allows us to relate a list of objects (e.g. genes) to the vocabulary of an ontology (e.g. GO terms) through a mapping matrix that assigns each object to the terms organized in a Directed Acyclic Graph (DAG) (e.g. the GO DAG) associated with the ontology. Once the usage of the graph theory concepts in terms of Carey's framework was described, we could focus on the semantic similarity measure proposed by Lord et al. This measure is based on the Information Content (IC) concept (see section). It was proposed by Resnik ([128]).

The literature offers a large list of measures for computing the semantic similarity between GO terms. Most of them rely on the length of the shortest path. These methods suggest that the closer two nodes are, the more similar they are. However, when dealing with the GO, ontology DAGs generally speaking, this strategy does not capture the meaning of the links between terms. Resnik suggested a measure that was based on the probability of appearance of a GO term, which is the IC. However, we detected that the properties associated with the definition suggested by Resnik, as well as other concepts proposed in the bioinformatics literature, sometimes show a lack of clarity. Thus, in a minor contribution we proved the monotonic property of the probability theory adapted to Carey's framework (see section 3.3). Moreover, we saw that the IC captures the importance of the meaning of a single term and the OOC revealed that the root is the most abstract term. Therefore, based on the monotonic property we proved that the root node of an ontology is the term with the lowest IC, which is in fact null.

Based on the IC, Resnik introduced a semantic similarity measure that depends on the Most Informative Common Ancestor (MICA) (se section 3.5.3). The idea is that the IC of GO terms relies on the relationships given by the DAG structure of the ontology. Thus, the shared information between two terms is usually proportional to the IC of the MICA in the rooted DAG. In this regard, Lord *et al.* ([98]) argued that such a measure only selects the one common ancestor, and they suggested an alternative measure that depends on the minimum probability of a term when there are more than one shared parents. However, in a major contribution of this thesis we proved that such a "new" measure of semantic similarity is in fact the same as the one proposed by Resnik ([128]).

With respect to the IC we suggested a proposition that allows us to compute the matrix with the number of times that each term or any of its specializations refer to a specific ancestor, and a corollary that allows us to compute the number of times that a term or any of its refinements appears in the OOC. Some parts of the proofs associated with these results were the clue when we developed some `R` functions of the package `sims` for computing semantic similarities. Moreover, based on the idea that a distance can be computed in terms of a similarity, we proved that by rewriting Resniks second measure as a distance, then it is in fact a metric distance (see section 4.2.2.2).

The pseudo-distances proposed by Joslyn et al. ([84], [85], [83]) are based on POSET theory as mentioned above. Joslyn's approach is a totally different strategy for four main reasons. First, it is an edge-based approach, which means that the measures are computed based on the topology of the DAG. Second, they are not semantic similarities, they are "distances". That is, the inverse idea of semantic similarities, which does not mean the mathematical concept of inverse function. Third, Lord's measure is based on the IC, which is a probabilistic point of view, and Joslyn's measures are an algebraic point of view. And fourth, pseudo-distances can only be computed between comparable terms, which are not the case with the Lord's measure.

When we focused our effort on pseudo-distance, we realized that Joslyn et al. defined a good mathematical framework. It is based on the POSET theory and suggests constructing a POSET Ontology (POSO) structure in order to establish the relation between the objects and terms mentioned above (see section 3.4.6). We highlighted that POSETs are general combinatorial structures basically equivalent to DAGs (see section 3.4). But, what is clear is that the is a certain level of analogy between the OOC and the POSO (see section 4.2.1). The concepts used in the definition of the POSO and in OOC can be easily linked. The poset is the vocabulary of the Ontology, the set of objects is the Object in the OOC, and the mapping function is the mapping between the Ontology and the list of objects in the OOC. Therefore, we emphasized that both structures were in fact two ways for formalizing mathematically "how to attribute biological meaning".

We saw that pseudo-distances are useful measures for computing how "different" two terms in a POSET are. However, these types of measures only make sense when we are considering comparable terms. But, when restricting ourselves to these comparable terms, a pseudo-distance is actually a distance, and by focusing on the minimum chain length measure we proved that it is a metric distance. In this regard, we highlighted that by assuming comparable terms we could prove that the other pseudo-distances are metric distances too 4.2.3.

The second specific objective was the development of an `R` package for computing semantic similarities between ontology terms. This package was called `sims`. It allows us to compute semantic similarities between terms in an arbitrary ontology. That is, it is not restricted to the GO. To do this, the key point is an object of class `S4` called `OOC`. It is merely used as a container for the ontology vocabulary, the refinement matrix associated with the ontology, the list of objects and the mapping matrix. Fourteen measures from different approaches were implemented. Specifically, from the node-based approach the seven semantic similarity measures proposed by Resnik ([128]), Lin ([97]), Schlicker *et al.* ([134]), Jiang and Conrath ([82]), Mazandu and Mulder ([107]), Pirró and Seco ([124]), and Pirró and Euzenat ([123]) were implemented. With regard to edge-based approaches the two semantic similarity measures proposed by Resnik ([128]), and Rada *et al.* ([127]) were implemented, as well as the distance measure proposed by Rada *et al.* ([127]) and the four initial pseudo-distances proposed by Joslyn *et al.* ([84], [85], [83]).

The package can manage any ontology as mentioned above, but it was also designed for proving specific functions devoted to dealing with the GO. In this regard, there are some functions that allow us to build the refinement matrix associated with the induced subgraph of the GO, the mapping matrix that assigns a list of Entrez Gene IDs to the GO IDs. In addition, due to the fact that a researcher sometimes needs to compare two lists of genes in terms of their biological annotation, the `sims` package has some functions intended to do this task. Given the two lists of genes, the idea relies on looking for all the GO terms where both lists of genes are annotated, then extracting the common induced subgraph from the GO DAG and computing the semantic similarities associated with each list. These facts allow us to build what we called the *semantic similarity profiles* (see section 4.2.5). Based on these profiles, the package allows us to compute some descriptive statistics associated with each profile, perform a Matel's Test and calculate the cosine similarity measure, as well as plotting different types of figures that help us to compare similarities or differences between both groups of semantic similarities profiles.

During the package development process, some functions that we were programming suggested algorithms that could be used for proving some basic properties of the graph theory. The most representative was the Handshaking Theorem and its corollary (see section 4.1.1). After surveying the literature ([155], [155]), ([17]) we observed that our ideas could be conceived as an alternative way to prove these properties. For this reason, we thought it appropriate to provide such formalisms as minor contributions (see section 4.1.1).

There are other packages for computing semantic similarities available on the Bioconductor project ([63]) website: the `GOSim` ([59]) and the `GOSemSim` ([165]). They are good packages for measuring semantic similarities between GO terms. However, the `sims` package goes one step further than these. As was mentioned above, it does not exclusively depend on the GO terms because we can compute semantic similarities between terms of an arbitrary ontology. Furthermore, while `GOSim` and `GOSemSim` provide five and four similarity measures based on node-based approaches, `sims` provide fourteen measures based on node-based and edge-based approaches. However, `GOSim` and `GOSemSim` offer the possibility of computing functional similarities, and `sims` package do not. That is, they can calculate a specific number based on a similarity measure for two lists of genes or two lists of GO terms. However, in our opinion this type of computation does not provide a complete biological understanding about the differences or similarities between the two lists of objects that are being compared. For this reason we thought it suitable to provide a statistical point of view, and so we introduced the semantic similarity profiles concept as mentioned above. Finally, we did not perform a formal comparison between the three packages. However, we observed that our package seems to perform computations more quickly than the other. We think that this might be due to the fact that our functions are based on matrix algebra and that they apply computational loops. In this regard, it would be interesting to mathematically or empirically study such a priori observed differences.

Natural extensions of this research might be divided in two main lines of work. On the one hand, the study that we have conducted on the relationship between the measures from node-based and edge-based approaches and the metric distance could be extended to hybrid-based approaches. Moreover, it would be interesting to try to find a theory that would unify the different approaches and allows you to switch from one type to another approach. On the other hand, we have observed that `sims` shows a very interesting behavior when it is performing the computations. `sims` seems to calculate semantic similarities very fast. We think that this may be because, after doing a quick inspection of the functions from the other packages seems to suggest that they have adopted a computer-based approach by considering loops, however, we have implemented the functions by applying a matrix approach. Neither an empirical study nor a theoretical proof based on the order of computations has been performed in order to validate our suspicion about the speed `sims`. So, obviously, it would be interesting to perform this task. However, unlike the other packages, `sims` does not provide normalized semantic similarity measures. Therefore, if a user wants to compare the results performed with different measures, and even to combine them, it could not be done at the current state of the package. Therefore, an extension in this direction would be very valuable.

Part III

Classification of GO Tools for Enrichment Analysis

Experimental omic technologies ([141]) have become both popular and affordable over the last decade, leading to a considerable increase in experiments and publicly available functional data sets. These high-throughput methodologies pose different challenges: the experiment itself, the statistical analysis of the data ([136]) and the obtaining of biological knowledge from the data ([144]). For example, in gene-expression studies, it is very common for the statistical analysis to yield long lists of genes and one of the main challenges is how to give these lists a biological interpretation ([131]). It might be reasonable to expect that this could be done relying on the information stored in the existing biological databases, which can help to relate the experimental results with previously existing biological knowledge.

The *Gene Ontology* (GO) has been presented, in section 1.2 of chapter 1, as a useful resource to provide biological interpretation and to answer the need of automation ([148]). The GO is a cooperative project, which was set in motion in the late 90s, developed and maintained by the GO Consortium ([148], [149], [150]). Briefly, it is an annotation database originated "to provide a controlled vocabulary to describe gene and gene product attributes in any organism". It consists of three ontology domains: *Biological Process* (BP), *Molecular Function* (MF) and *Cellular Component* (CC). Each of them is represented as a Directed Acyclic Graph (DAG) ([43]) with two kinds of relationships (*is-a* and *part-of*) and whose nodes are the GO terms arranged from the most specific ones at the bottom to the top which is the most general term. The gene products may be linked to one or more GO terms in these ontologies. Thus, when a given gene has been annotated to a GO term it is also linked to its related nodes.

In recent years, many tools have been developed to assist with the analysis of experimental results based on the GO. Some of these tools are intended to manage functional annotations while others are specific for analyzing gene lists and many allow both possibilities ([89]). The scientific community has rapidly moved from lacking the appropriate GO tools to having a wide range of applications with, apparently, very similar capabilities. It seems reasonable to ask ourselves whether it is worthwhile to keep developing new variants of the same programs. We may have reached the point where most of the needs might be solved by already existing tools and the question has simply shifted to decide which of these tools, among those available, is the one that best fits the objectives that are being pursued.

This part addresses the second main objective of this thesis in 2.2.1. That is, “to classify and study the evolution of GO tools for enrichment analysis”. To carry out this classification and study of GO tools, specific objectives stated in section 2.2.2.1 are answered one by one. Thus, in chapter 6 material and methods associated with specific objectives are introduced or described, in chapter 7 results associated with specific objectives are presented, and finally, in chapter 8 results are discussed.

Chapter 6

Material and Methods

This chapter introduces and describes materials and methods used to answer the five specific objectives of this second part of the thesis in 2.2.2.1. Thus, each main section of the chapter is associated with one of the specific objectives. Briefly, in order to establish a list of standard functionalities for classifying the reviewed tools, the first section explains how the selection process of tools for enrichment analysis based on the GO was conducted. The second section describes the criteria used to define an Standard Functionalities Set, and how the tools reviewed were classified based on their capabilities and according to this set of standard features. Based on this set of standard functionalities, the third section is intended to describe how a web-tool called `SerbGO`, devoted to both searching for and comparing GO tools, was designed and implemented. The fourth section presents the statistical methods that have been used to study the evolution of the original GO tools, which were classified and stored in the `SerbGO` database in order to identify models or patterns of tools according to their capabilities. Finally, the fifth section presents basic concepts and principles of ontologies and how they have been used to develop an ontology called `DeGOT`.

6.1 Selection of GO Tools

A long list of applications available at the GO website was reviewed from the existing literature ([148]). Due to the high heterogeneity among different types of tools it was decided to focus only on *Tools for Gene Expression/Microarray Analysis* (`http://www.geneontology.org/GO.tools.microarray.shtml`).

These tools use either the ontologies or the gene associations provided by the GO Consortium to facilitate the analysis of gene expression data. It must be noted that the presence in the GO website does not imply approval by *The GO Consortium* ([152]) and does not mean these tools have been either tested or found to use the information accurately. As the GO Consortium staff says, this list "is provided to promote an exchange of information between users and software developers".

The list of applications which was finally included with the associated promoter entity and references is provided in table 6.1.

GO Tool	Promoter Entity	Reference
CLENCH	Huck Institutes of the Life Sciences, Penn State	[137]
DAVID	National Institute of Allergy and Infectious Diseases	[40]
EASE		[40]
eGOn	Norwegian University of Science and Technology and Norwegian Microarray Consortium	[116], [13]
ErmineJ	Center for Computational Biology and Bioinformatics, Columbia University	-
FatiGO	Bioinformatics Department at the Centro de Investigacion Principe Felipe (Spain)	[3]
FuncAssociate	Roth Computational Biology Laboratory, Harvard Medical School	[16]
GARBAN	University of Navarra, Spain	[105]
GeneMerge	Harvard University	[26]
GFINDer	Bio-Medical Informatics Laboratory at the Politecnico di Milano	[106]
GOArray	Yale Center for Medical Informatics	-
GoMiner	Genomics and Bioinformatics Group of LMP, NCI, NIH	[166]
MatchMiner	and Medical Informatics and Bioimaging group of BME, Georgia Tech/Emory University	[21]
GOstat	Walter and Eliza Hall Institute of Medical Research, Melbourne, Australia	[12]
GoSurfer	Harvard School of Public Health	[169]
GOTM	University of Tennessee Genome Science and Technology and Oak Ridge National Laboratory (ORNL)	[168]
GOToolBox	Developmental Biology Institute of Marseille	[104]
MAPPFinder	Gladstone Institutes, University of California	[44]
NetAffx	Affymetrix	[29]
Onto-Tools	Intelligent Systems and Bioinformatics Laboratory, Wayne State University	[46],[88]
OntoGate(OntoBlast)	Max Planck Institute for Molecular Genetics	[167]
Ontologizer	Charité University Hospital, Germany	[129]
Ontology Traverser	Baylor College of Medicine	[164]
SeqExpress	SeqExpress	[19]
SOURCE	Stanford Microarray Database	[42]
THEA	Virtual Biology Lab at the Institute of Signaling, Developmental Biology and Cancer Research	[120]

Table 6.1: GO tools selected for defining the and Standard Functionalities Set.

Most of these programs are constituted by only one platform. However, some of them are added in different tools (e.g. `GoMiner` and `MatchMiner` or `DAVID` and `EASE`). According to this, it must be advised that in order to facilitate the review and the selection process, five different programs, namely, `Onto-Compare`, `Onto-Design`, `Onto-Express`, `Onto-Miner` and `Onto-Translate`, whose promoter is the *Intelligent Systems and Bioinformatics Laboratory*, were considered as only one platform that has been called Onto-Tools.

6.2 Definition of Standard Functionalities Set and Classification of GO Tools

The review yielded a substantial number of heterogeneous features, which were grouped into a potential set of functionalities. After several iterations, the features initially selected were converted into specific functionalities once redundancies were excluded. This process resulted in a set of features arranged in 205 standard functionalities.

Capabilities of each GO tool analyzed were classified *in situ* according to the Standard Functionalities Set and by taking the following criteria into account:

1. The functionality was available in the GO tool.

2. The functionality was mentioned in the publication but it could not be validated.

3. The functionality was neither found in the paper nor in the application.

6.3 SerbGO: Searching for the Best GO Tool

SerbGO is a web-based application designed to:

1. facilitate for researchers the task of determining which of the existing tools are appropriate for their needs and

2. to enable a comparison between some of the available tools.

Figure 6.1 shows the workflow to perform both actions of analysis.

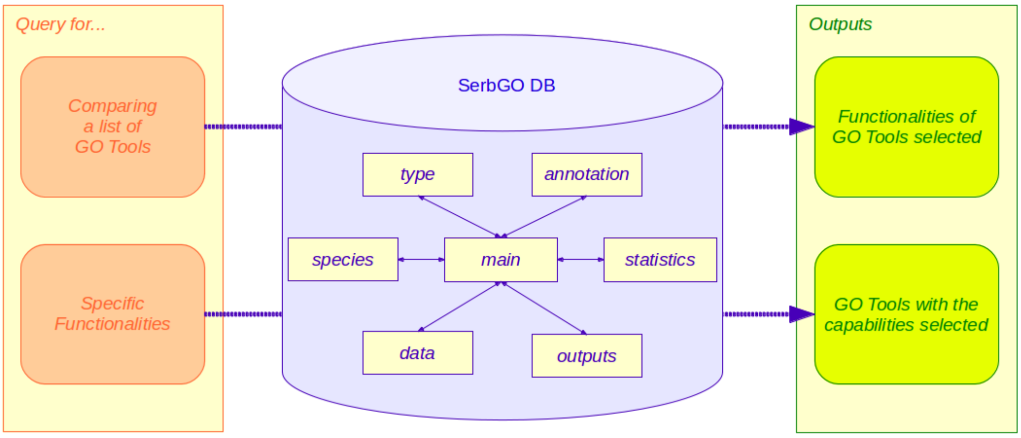

Query for...
Comparing a list of GO Tools
Specific Functionalities
SerbGO DB
type
annotation
species
main
statistics
data
outputs
Outputs
Functionalities of GO Tools selected
GO Tools with the capabilities selected

Figure 6.1: SerbGO workflow

6.3.1 Implementation of SerbGO

SerbGO is a web tool developed in PHP ([1]) using the ADOdb Database Abstraction Library for PHP ([96]) and the Javascript language ([108]) to increase its interactivity. It works accurately on most of the web browsers. It has been adapted and validated specifically for Mozilla Firefox, Internet Explorer, Konqueror, Chromium and Opera web browsers.

Information about tools and their functionalities have been stored in a database implemented in the open source relational database management system MySQL ([76]).

6.4 Evolution and Clustering of GO Tools

Periodically, but not regularly, the SerbGO database is reviewed. Such reviews consist of three basic steps: removing tools that are no longer available, updating the classification of existing tools when promoters modify their capabilities, and appending records with new tools available at the GO Consortium website. This fact led us to observe a certain degree of evolution in tools classified in the SerbGO database. For this reason we decided to conduct a statistical study, based on the monitoring of all the tools included in the first version of SerbGO, in order to discuss the evolution of the functionalities, and to observe if some form of clustering of GO tools exists.

The following subsections describe the data and the statistical methods that have been used to perform such a statistical analysis.

6.4.1 Data

6.4.1.1 Raw Data

Data analysis is based on the *number of standard functionalities* that 26 tools had available in 3 different years (2005, 2007 and 2009). These tools are the list of the original GO tools stored in the first SerbGO database. For list of GO tools, six of the tables (*type*, *species*, *data*, *annotation*, *statistics* and *outputs*) stored in SerbGO database (see section 7.3.2), corresponding to each year, were downloaded.

6.4.1.2 Homogenization of Raw Data

After downloading information from SerbGO, an homogenization process of raw data has been performed in order to reduce redundancies. This process consists of three steps:

1. *To homogenize some field names from 2005 tables.* During the submission process and after publication of SerbGO, it was necessary to introduce some modifications in the PHP code and the structure of the database. Due to this fact some field names were modified.

2. *To relabel functionalities mentioned in references but not validated.* It does happen sometimes that a functionality is mentioned in the reference reviewed, but it is not possible to validate it *in situ* (see section 6.2). When this situation occurs, a number 9 is stored in the associated record instead of annotating either a number 1 (i.e. functionality present) or a number 0 (i.e. functionality absent). Thus, since this analysis prefer to be conservative, values 9 have been relabeled, as missing functionalities, that is, as values 0.

3. *To reduce functionalities that are too specific, which by themselves do not provide extra information and would add redundancy to the data.* Some functionalities from different tables (see section 6.3.1) are highly specific. Actually, they are rarely used for an specific analysis, and have remained as a mere property of SerbGO. Thus, these characteristics have been removed from the tables because their major immediate functionalities already hold the relevant information.

The homogenization of these tables resulted in a reduction of the number of functionalities. From the original 205 standard functionalities, 178 functionalities have been selected and pre-processed to be analyzed.

6.4.1.3 Data Matrices

For each year, the seven homogenized tables were merged into one unique matrix of large data. That is, data analysis has been based on three binary matrices[1] such that each matrix describes the capabilities of the GO tools under study through the functionalities selected for a specific year (i.e. 2005, 2007 or 2009). Formally, that is:

$$[H]\ \begin{array}{c|cccc|} & \text{Func. 1} & \text{Func. 2} & \dots & \text{Func. m} \\ \text{GO tool 1} & x_{11} & x_{12} & \cdots & x_{1m} \\ \text{GO tool 2} & x_{21} & x_{22} & \cdots & x_{2m} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \text{GO tool n} & x_{n1} & x_{n2} & \cdots & x_{nm} \end{array} = \mathbf{X}_{year}$$

where

$$x_{ij} = \begin{cases} 1 & \text{GO tool } i \text{ has the functionality } j \\ 0 & \text{otherwise.} \end{cases} \tag{6.1}$$

6.4.2 Statistical Methods

The statistical analysis to study the evolution and clustering of functionalities of GO tools has consisted of descriptive statistics, and inferential and a multivariate analyses. The following subsections describe the methods used in each part of the analysis.

The data analysis has been performed with the statistical software `R` ([126]) supported by some extra packages, and explicitly programmed functions. These functions and raw data can be downloaded from the `GitHub` repository `https://github.com/jlmosquera/gotoolsevolution`.

6.4.2.1 Descriptive Statistics Methods

Descriptive statistics have been performed to provide basic summaries about samples and observations. It may be used to describe relationships between pairs of variables. In this case, descriptive statistics include:

[1] A *binary matrix* (*aka* zero-one matrix or boolean matrix) is an integer matrix in which each element is 0 or 1.

- *Contingency tables*: to show frequencies of the number of standard functionalities present in each GO tool for each year. These tables are presented globally and disaggregated by section. Each contingency table is accompanied by its bar diagram.
- *Bar diagrams*: to display in the same plot the distributions of the functionalities associated with each year.

6.4.2.2 Inferential Analysis Methods

Inferential analysis has been used to test hypotheses such as whether or not differences exist between frequencies of functionalities from different years. To be more specific, both globally and through disaggregating by sections, the following inferential methods were performed:

- *Chi-Squared Tests of Homogeneity* ([115]): to test whether frequency counts of GO tools are distributed identically across different years. These tests were accompanied by the following descriptive plots:
  - *Boxplots*: to display graphically the number of functionalities of GO tools per year.
  - *Scatterplots*: GO tools are displayed as a collection of points in a diagram of Cartesian coordinates, where values of each axis are the frequencies of functionalities associated with each GO tool per each pair of years.
- *Locally Weighted Regression* (*Loess*): to provide a graphical summary of the relationship between frequencies of functionalities available in GO tools for two different years, smooth curves and their confidence bands were fitted to scatterplots.

Loess is one of the most popular smoothing methods to model a relation between an explanatory variable and a response variable, by locally fitting a polynomial curve ([30], [31], [33]). Here, loess curves were fitted to explain the number of functionalities available in GO tools in one year based on the number of functionalities available from a previous year. That is, the number of functionalities available in each GO tool were represented in scatterplots for each pair of years (i.e. 2005 *vs.* 2007, 2007 *vs.* 2009, and 2005 *vs.* 2009) and for each of them a Loess curve and their confidence bands were fitted in order to model the evolution of GO tool capabilities from one year to a later on.

Loess has been performed using the function `loess` from the `R` package `stats` ([126]). This function is based on the method purposed by Cleveland *et al.* ([32]).

6.4.2.3 Multivariate Analysis Methods

Multivariate analysis has been performed in order to explore the behavior of GO tools according to their capabilities over time. That is, multivariate methods have been applied to study how the different programs are grouped according to their capabilities throughout the years. The multivariate methods that have been applied in the analysis are:

- *Hierarchical Clustering*: to identify groups of GO tools.
- *Multidimensional Scalings* (*MDS*): to obtain spatial representations in reduced dimensions of each dissimilarity matrix and help with the task of identifying potential clusters.
- *Mantel Test*: to study the association among the three dissimilarity matrices.

The following paragraphs introduce and describe some basic concepts of these multivariate methods in short.

Multivariate analysis has been performed using different `R` ([126]) functions from packages `fpc` ([75]), `MASS` ([156]), `rgl` ([2]), `stats` ([126]), and `vegan` ([118]), and some specific functions have been developed for this analysis.

Similarity Coefficients and Dissimilarity Matrices

For each initial data matrix $\mathbf{X}_{year}$ (see section 6.4.1.3) two different similarity measures have been used afterwards to compute their associated dissimilarity matrices. The first is the *Jaccard Coefficient* ([81]), and the second is the *Matching Coefficient* ([143]). Dissimilarities were derived from similarities using the formula suggested by Cuadras ([38]) (see section 3.5.1). These similarity measures have been calculated based on the counts of matches (i.e. 1) and mismatches (i.e. 0) for each functionality and each pair GO tools i and j (see table 6.2).

Dissimilarity matrices based on the Jaccard coefficient have been computed using the `R` function `vegdist` from the package `vegan` ([118]), and dissimilarity matrices based on the Matching coefficient have been explicitly programmed.

		GO tool j		
		1	**0**	**Total**
GO Tool i	**1**	a	b	$a + b$
	0	c	d	$c + d$
	Total	$a + c$	$b + d$	$m = a + b + c + d$

Table 6.2: Contingency table displaying frequencies for each particular combination of matches (i.e. 1) and missmatches (i.e. 0) of two arbitrary GO tools.

Hierarchical Clustering

An *agglomerative hierarchical clustering* ([103], [52], [53], [73]) for each dissimilarity matrix has been performed. The *metric distances* used have been based on the Jaccard and Matching coefficients as mentioned in previous section 6.4.2.3, and *Average Link* (*aka* UPGMA) has been used as criterion of *clustering method* ([143]).

In order to visualize similarities between GO tools and to try to identify how they are grouped, for each dissimilarity matrix a *dendrogram* ([103], [52], [53], [73]) has been plotted.

Hierachical clustering analysis has been performed with the `R` function `hclust` from the package `stats` ([126]).

Determination of the Number of Clusters

In order to identify the optimal number of clusters, *Silhouette plots* ([53], [87]) based on the non-hierarchical method of *Partitioning Around Medoids* (*PAM*) ([53], [87]) have been applied.

The PAM algorithm has been run several times for different numbers of clusters, in order to compare the results and determine what the "optimal" number of clusters is. This has been done by using the `R` function `pam` from the package `fpc` ([75]). This function was not in fact executed directly, but was included in out own function written for this analysis.

Average Silhouette Widths for the entire data sets ([53],[87]) have been used for calculating the *Silhouette Coefficients* (*SC*) ([87]), and determining what the "optimal" number of clusters should be. Table 6.3 gives some hints for interpreting the *SC* values ([87]).

Range of SC	Interpretation
$(0.7, 1.0]$	A strong cluster structure.
$(0.5, 0.7]$	A reasonable cluster structure.
$(0.2, 0.5]$	A weak or artificial cluster structure.
≤ 0.25	Lack of substantial cluster structure.

Table 6.3: Interpretation of Silhouette Coefficient.

Specific functions have been coded for generating bar plots of the Average Silhouette Widths for the entire data set with respect to the number of clusters. In each of these plots a red point indicates the Silhouette Coefficient.

Multidimensional Scaling

Two different *Multidimensional Scaling* (MDS) ([36], [18], [20], [18]) strategies have been performed for each dissimilarity matrix 6.4.2.3: a metric MDS based on the *Classical Multidimensional Scaling* ([153], [154], [65]), and a non-metric MDS based on *Kruskal's Non-metric Multidimensional Scaling* ([91], [138], [139]).

Classical MDS's have been performed using the `R` function `cmdscale` from the package `stats` ([126]), and Kruskal's Non-metric MDS have been performed by using the `R` function `isoMDS` from the package `MASS` ([156]).

In order to assess the *Adequacy of Dimensionality* ([53], [52]), the *agreement measure* P_m^2 proposed by Mardia *et al.*[2] ([103]) has been calculated for each dimension of each classical MDS solution performed. They are shown as percentages in labels of $x-$ and $y-$s axes of scatter plots associated with the MDS solutions. But also, in order to show what the "explained variability"

[2] The *agrrement measure* P_m^2 proposed by Mardia *et al.* is defined as

$$P_m^2 = \frac{\sum_{i=1}^{m} \lambda_i^2}{\sum_{i=1}^{n-1} \lambda_i^2}$$

where λ_i are the eigenvalues associated with the matrix $\mathbf{B} = \mathbf{X}\mathbf{X}^{\mathbf{T}}$, and where $\mathbf{X}_{n\times m}$ is the matrix of the coordinate axes of the MDS space.

(i.e. the accumulated measure of agreement) is for each component of each classical MDS solution, bar diagrams displaying the measures of agreement P_m^2 with respect to the number of dimensions have been plotted.

Stress-1[3] proposed by Kruskal ([91]) for measuring the lack of fit between ideal distances and fitted distances has been calculated for each resulting solution from Kruskal's Non-metric MDS performed. Table 6.4 shows some hints suggested by Kruskal ([91]) for the interpretation of stress.

Stress (%)	Goodness of fit
> 0.20	Poor
0.100	Fair
0.050	Good
0.025	Excellent
0	Perfect

Table 6.4: Stress and goodness of fit of the MDS solution based on the guidelines suggested by Kruskal ([91]).

However, due to the fact that stress measure has been the object of some criticism ([160], [18]), *Scree plots* and *Shepard diagrams* ([18], [69]) have been plotted too.

Specific functions have been programmed for computing Adequacy and stress $\sigma_1(\mathbf{X})$ and yielding Scree plot and Shepard diagram for each MDS solution.

Mantel Test

In order to test the correlation between dissimilarity matrices, two tests have been applied[4]: the simple *Mantel Test* ([101], [102]), and the *Partial*

[3] The *Stress-1* proposed by Kruskal ([91]) is defined as

$$\sigma_1(\mathbf{X}) = \sqrt{\frac{\sum_{i \neq j = 1 \ldots n} (f(p_{ij}) - d_{ij}(\mathbf{X}))^2}{\sum_{i \neq j = 1 \ldots n} d_{ij}^2(\mathbf{X})}}$$

where $f(p_{ij})$ are the ideal distances in the m-dimensional space $\mathbf{X}$, and $d_{ij}(\mathbf{X})$ are the fitted distances.

[4] These tests might of course be placed in the Inferential Analysis Methods 6.4.2.2, but we decided to include these methods in this section because dissimilarity matrices are involved.

Mantel Test ([142], [92], [93], [100]). The first has been performed for testing the correlation between each pair of dissimilarity matrices, and the second has been performed for testing the correlation between two dissimilarity matrices, while controlling the effect of the third dissimilarity matrix, in order to remove spurious correlations[5].

The Mantel Test and Partial Mantel Test have been performed with the `R` functions `mantel` and `mantel.partial`, respectively, from the package `vegan` ([118]).

6.5 `DeGOT`: An Ontology for Developing GO Tools

When a researcher needs to give a biological interpretation to the results yielded in an omic experiment, she/he often makes use of the information stored in the GO 1.2. In section 6.1, it was mentioned that a large quantity of methods and tools for mining and managing information stored in the GO, have been developed during the last decade. In this sense, the definition of a Standard Functionalities Set 6.2, and subsequent classification of GO tools based on their capabilities, and according to this *Standard Functionalities Set* was required. This fact allowed us to build the `SerbGO` database and its associated web application to query GO tools that more or less fit the goals that are being pursued by a researcher, or who wants to compare capabilities of different programs.

In next chapter 7 below, we will see that results of data analysis about the evolution of GO tools (see section 7.4) suggest that most of the needs associated with the traditional enrichment analysis are covered. However, many developers are still working on GO tools, either by improving them or by programming new ones.

Developing a new program is not an easy task. Many factors must be taken into account and many relevant questions must be answered during the period of program design. For example,

- What type of user is the tool intended for?

[5] Mantel developed an asymptotic test, but we have used permutations. There are different permutation procedures [92]. In this analysis the method applied is based on permuting the objects in the first matrix so that the correlation structure between second and first matrices is kept constant ([92]).

- For what types of questions should the information stored provide answers?
- What are the species that the tool will cover? Should the tool actually be independent from the species?
- Which statistical methods are going to be used?
- Who will curate the tool?
- What kind of structure of information is going to be supported in this resource?
- What kinds of inputs and outputs are going to be required?
- ...

As a starting point, SerbGO is a good tool to answer some of these questions and so help developers have an idea about what kinds of functionalities are “usually” implemented. However, the dynamism of the needs of potential users and improvements in methodologies, make the design of a new GO tool a necessary task. Therefore, in order to help with these kinds of questions, an ontology called DeGOT (for *Developing GO Tools*) has been built.

DeGOT is an ontology intended to help developers build a new GO tool, as well as to perform more complex searches than in SerbGO. The reason for building an ontology is that in a relational database, such as SerbGO, concepts are stored using tables, however the system does not contain any information about what these concepts mean and how they relate to each other. Ontologies do provide the means to store such information, which allows for a much richer way to store information. This also means that a user/developer can perform fairly complex and advanced queries.

Ontologies are one of the backbones of the Semantic Web ([14], [74]), although they do not have an standard definition. There are many ways of defining and developing ontologies ([67], [68]). For the purposes of this thesis, and for a better understanding of how DeGOT has been implemented and developed, basic concepts about ontologies and semantic webs are introduced in the following subsections.

6.5.1 Basic Concepts of an Ontology

In general, ontologies are used for capturing conceptualizations of knowledge domains and then facilitate both communication between users or researchers and the usage of domain knowledge by computers for multiple purposes.

6.5.2 Domain of Knowledge

Formally, a *Domain of Knowledge* is a form of cognizance used to refer to an area of human effort, an autonomous computer activity, or another specialized discipline ([77]). For instance, the GO project provides ontologies to describe attributes of gene products in three nonoverlapping domains of molecular biology knowledge: Molecular Function (MF), Biological Process (BP), and Cellular Component (CC).

6.5.3 Structure and Constructs of an Ontology

The way that an *ontology* tries to describe the concepts in a domain of knowledge, as well as the relationships that exist between those concepts, is by providing formal explicit descriptions. These formalizations are based on three main components or components of information: classes, properties and role restrictions. Without going into details:

- *classes* of objects denote concepts of the domain (sometimes they are also called *concepts*),
- *properties* (*aka roles* or *slots*) describe features and attributes of each concept, that is they are the relationships, and
- *role restrictions* (or *facets*) are restrictions on properties.

For example, in the GO, the term *Signal Transduction* denotes the concept of *The cellular process in which a signal is conveyed to trigger a change in the activity or state of a cell.* It has a property that consists in *Cell Surface Receptor Signalling Pathway* ***is a*** *Signal Transduction.* But, there is a restriction and it is that ***only*** *Regulation of Signal Transduction* ***regulates*** *the Signal Transduction.*

There is a last component called *individuals* that are instances of a class. They are not necessarily defined in an ontology structure. For example, GO, like many ontologies, does not use instances. Terms in GO represent a class of entities or phenomena, rather than specific manifestations thereof. The reason is mentioned in subsection 6.5.2, the GO *describe attributes of gene products*, but genes are not individuals of these, even, of course, if they have been linked in terms. An individual, ontologically speaking, is a specific example of something. For example, a molecular biologist is a scientis, but James Watson and Francis Crick are two different instances of a molecular biologist, rather than a subtype of molecular biologist. However, if we know that a molecular biologist is a scientist, then it can be said that every instance of a molecular biologist is a scientist. Thus, an ontology, together with a set of individuals of classes, constitutes a Knowledge Base[6].

6.5.4 Basic Concepts of a Semantic Web

Nowadays, most of the web content generated is intended for human comprehension, but not for computers. Lets see a couple of examples. First, when the web content is provided automatically by a computer, the information is often presented without an appropriate structure that permits us to perform other processes later on. Second, when a user tries to extract information, usually, he/she has to fill out forms. However, such formularies are sometimes designed to access the information, with the exception of keyword-based searches (e.g. `Google`, `Yahoo`,...). But, even in these cases, keyword-based searches show several drawbacks. For instance, results are highly sensitive to vocabulary, human involvement is necessary to interpret and combine results and web searches are not readily accessible by other software tools. In summary, the meaning of Web content is not machine-accessible, there is a lack of semantics ([74]). An approach to solve this situation is the Semantic Web.

A *Semantic Web* (or vocabulary) is a collection of URIs[7] with described meaning or a special type of ontology. That is, a Semantic Web represents the web content in a much easier way to be processed by machines, and allows the use of intelligent methods to take advantage of these representations.

6.5.4.1 Semantic Web Languages

Most of the web content is programmed for human readers rather than software tools as mentioned above. In order to reduce the lack of processing

[6] A *knowledge Base* (*KB*) is a technology used to store complex structured and unstructured information used by a computer system ([95]).

[7] *Uniform Resource Identifiers* (URIs) ([74]) are a way of identifying resources, specially but not exclusively on the Web.

capabilities having software tools, the scientific community has been developing semantic web languages that allow both humans and machines to read, interpret and process the data.

The most relevant semantic web languages are briefly described in the following paragraphs.

HTML: HyperText Markup Language

HyperText Markup Language (HTML) is a standard markup language used to create web pages ([66]). This language is written in the form of elements consisting of tags (vocabulary). Thus, when a web browser reads an HTML document, then it does not display the tags, but uses these tags to interpret the content of the page.

HTML was originally conceived as a semantic language free of presentation details. However, some browser vendors developed attributes and elements devoted to improve the presentation of the web contents. It is hardly surprising that web pages are predominantly written in HTML. Thus, humans have no problems in reading and interpreting HTML documents, but machines cannot process the data in these types of files. A better representation of written data is XML.

XML: Extensible Markup Language

Extensible Markup Language (XML) is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable ([7]).

XML is not about displaying data. It is different from HTML because developers must create their own tags to annotate data. XML is a flexible text format language that is used to structure, store, and share information. However, XML does not provide any method of talking about the meaning of data, that is, the semantics of data.

RDF: Resource Description Framework

Resource Description Framework (RDF) is a standard for describing resources on the web ([5]). It attributes meaning by encoding data in sets of triples, which are subject, predicate and object statements. Each element of a triple is identified by an URI, and URIs represent both resources and relations. It is important to highlight that RDF is a way of working with triples, not the file formats. In fact, it is to the semantic web what HTML is to the Web.

RDF is written in XML.

OWL: Web Ontologies Language

Web Ontology Language (OWL) ([6]) is a semantic web language designed for use by software tools that need to process and represent complex information about things, groups of things, and relations between things, instead of presenting information to humans. In other words, OWL can be thought of as an object-oriented language that defines classes, hierarchy of classes, attributes and relations, and it serves to implement ontologies for the web.

OWL is written in XML and it is more expressive than RDF.

6.5.5 Implementation of DeGOT

DeGOT is an ontology written in OWL by using the Protégé resource version 4.3 ([117]), which is supported by grant GM10331601 from the National Institute of General Medical Sciences of the United States National Institutes of Health.

Protégé is a free, open-source platform to build domain models and knowledge-based applications with ontologies. It is in fact the leading ontological tool ([117]).

Chapter 7

Results

7.1 Definition of the Standard Functionalities Set for Classifying GO Tools

The *Standard Functionalities Set* was organized in nine sections. Table 7.1 shows the number of standard functionalities per section.

Section	Num. of Functionalities
Tools for	2
Type of experiment	7
Interface	7
Availability	4
Supported species	26
Data	40
Annotation	70
Statistical analysis	26
Output	23

Table 7.1: Number of standard functionalities per section.

The following subsections describe the functionalities associated with each section of functionalities defined.

7.1.1 Tools for

There are three types of tools identified. These programs can be used for exploring GO data and/or annotating information associated with functional information. Thus, two functionalities for classifying the three groups of tools were defined. The first group is those tools focused on *Exploration.* These programs consists of a combination of statistical methods and functional annotation with some kind of graphical representation. The second group is those tools called *Annotation*, and they provide query-based access to functional annotation and produce tabular outputs. The third group is those tools that consist of both capabilities Exploration and Annotation.

7.1.2 Type of experiment

These programs are focused on data yielded with different types of high-throughput technologies and experiments. Based on the literature of the selected GO tools, five main types of platforms/experiments were identified. Most of the GO tools are devoted to dealing with data generated with *DNA Microarrays* focussed on two main types of platforms, *Spotted arrays* (e.g. cDNA-Chip) and *In situ arrays* (e.g. GeneChip of Affymetrix) ([132], [133]). But they deal with *Proteomics* data generated with high-throughput devices ([109], [27]), *SAGE* data from *Serial Analysis of Gene Expression* experiments ([130]), and *CGH* data from *Comparative Genomic Hybridization* experiments ([159]). Few of them deal with data generated with *Other* types of experiments or high-throughput technologies.

Table 7.2 shows the types of experiments defined.

Type of technologies and experiments
DNA Microarrays
Spotted arrays
In situ arrays
Proteomics
SAGE
CGH
Other

Table 7.2: Types of experiments and technologies defined.

7.1.3 Interface

Programs are available in one or more work interfaces and supported under one or more Operative Systems. Four categories were defined: *Web-based* when a program is available on-line, *Downloadable* if it is a GO tool that can be used locally after downloading it, *Command line* when a software can be managed in batch navigation, and *Compatible OS* indicating if the GO tools are available for Windows, Mac OS, Unix or Linux.

7.1.4 Availability

The GO tools can be used under different licenses. Three main types of licenses were identified and considered for classifying the programs: *Freeware* when a software is available without cost, *Partial Freeware* when a tool is intended for non-profit uses (usually public or academic users), and *Commercial* when it is a tool with some kind of cost for all users.

Many GO tools give the option of performing illustrative analysis with a set of samples. For this reason the *Sample file/sets* feature was considered as a capability of this section.

7.1.5 Supported species

Programs were classified according to the list of species available on the literature. Table 7.3 shows the list of species defined, which is ordered alphabetically.

Species Supported	
Anopheles gambiae	Arabidopsis thaliana
Ames Bos taurus	Bacillus anthracis
Caenorlabditis briggsae	Caenorlabditis elegans
Coxiella burnetii RSA 493	Danio rerio
Dictyostellium discoideum	Drosophila melanogaster
Fugu rubripes	Geobacter sulfurreducens PCA
Glossina morsitans	Homo sapiens
Leishmania major	Mus musculus
Oryza sativa	Plasmodium falciparum
Pseudomonas syringae DC300	Rattus norvegicus
Saccharomyces cervisiae	Schizosaccharomyces pombe
Shewanella oneidensis	Trypanosoma brucei
Vibrio cholerae	Other

Table 7.3: Supported species identified from the reviewed literature of the GO tools.

7.1.6 Data

Data can be grouped depending on whether the tools have *Automated Updating Sources.* If so, the periodicity for updating tool sources is classified as *Weekly*, *Monthly* or *Quarterly*.

To perform the analysis GO tools require some *Inputs* to access their sources. These input information is characterized as *Single Term*, *Evidence Codes* and other *Descriptions*. Inputs can be provided as *A List Of* identifiers that refer to *Genes*, *Proteins* or *Terms*. GO tools allow the loading of inputs from a *File* and/or by *Pasting Into A Text Area*.

Table 7.4 shows the common types of input data as well as the most widely used input *Identifiers*, which are listed alphabetically.

Automated Updating Sources		
Weekly		
Monthly		
Quarterly		
Inputs		
Single Term		
Evidence Codes		
Descriptions		
A List Of		
Genes		
Proteins		
Terms		
Load Inputs From		
File		
Paste Into A Text Area		
Input Identifiers		
Affymetrix ID	Clone ID	Chromosome Locations
Ensembl ID	Entrez ID	FISH clone ID
FlyBase ID	GenBank Accession Number	Gene Names
GenePept Accession	Gene Symbol	GO Consortium ID
GI Accession	HUGO Gene Names	IMAGE Clone
Entrez ID	PIR Accesion	Protein Accession
PubMed	RefSeq ID	Swiss-Prot ID
Symbol	Synonyms	TIGR-CMR
UniGene		
Cluster ID		
Names		
Symbol		
Additional Input Data	Other	

Table 7.4: Common input data options.

7.1.7 Annotation

Tools with both annotation and exploration functions provide a wide variety of options in the references evaluated. Three main groups of features have been organized in order to classify GO tools by their capabilities. These groups of characteristics are *Data Sources for Information Retrieval*, *Functional Annotations* and *Possibilities.*

Data Sources for Information Retrieval consist of a list of databases, tools or other types of resources. Functional Annotations set up a list of what kind of functional information is available in the program and/or give access. Possibilities is a list of information managements capabilities that some GO tools can apply.

Table 7.5 shows the lists of selected data sources, the main capabilities for managing information allowed by the software and features selected for classifying the GO tools based on their functional annotations.

7.1.8 Statistical Analysis

GO tools offer different statistical analyses depending on the focus, that is, annotation analysis, enrichment analysis or other analysis. The most common methods have been organized into four categories: *Analysis For*, *Test Used For Analysis*, *Correction For Multiple Tests*, and *Alternative Methods.*

Analysis for is devoted to classifying GO tools according to the features of the input datasets. There are three types of ways to provide the required lists of features: *A single list*, *A Query List vs a Reference List*, or *Multiple Lists.*

The most common approaches for assessing the significance of GO terms, sometimes called refinement of GO terms analysis, are the Single or Modular *Enrichment Analysis of GO terms* (see section 1.3.2 in chapter 1). This capability usually distinguishes if the analysis should be focused on *Under-represented* and/or *Over-represented* genes in the GO categories. The resulting significance of GO terms provided by the programs can usually be controlled by selecting a cutoff for raw *p-values* and/or *q-values* associated with different types of corrections for multiple testing.

The main *Tests Used For The Analysis* are the *Binomial*, *Chi-square*, *Hypergeometric*, *Fisher's Exact Test*, *Permutation Test* and *Other* tests.

Most of the GO tools offer the possibility of performing a *Correction For Multiple Testing.* Common methods applied are *Bonferroni or Modified Bonferroni* or *False Discovery Rate* that distinguishes between either *Benjamini–Hochberg* when assuming independence or *Benjamini–Yekutieli* when droppring independence, and *Family Wise Error Rate* that allows

Data Source or Field for Information Retrieval	
BioCarta	dbEST
EMBL	Ensembl
Entrez Gene	FlyBase
GenBank	GeneCards
GeneMap99	Gene Ontology
Gene Reference Into Function	HomoloGene
Human Genome Database	InterPro
KEGG	Map Location
Mouse Genome Database	Mouse Genome Informatics
NetAffx	OMIM
PDB	PFAM
PIR/Iprot	PlasmoDB
PubMed	Rat
Genome Database	RefSeq
RHdb	SGD
TAIR	TIGR
TrEMBL	UCSC
UniGene	UniProt
UniSTS	WormBase
Own Database	Other
Functional Annotations	
Biological pathways	Disease
Functional categories	General annotations
Literature	Protein domains
Protein interactions	Gene Ontology
	Molecular Functions
	Biological Processes
	Cellular Components
Possibilities	
Application Program Interface	Ordered-Input mode
File Import/Export	Integration with R
Append	Retrieval
Remove	Clustering options
Preprocess to Obtain IDs	Data transformations
Assess Bias	Reduce Redundancy
GO Ferms Filtering Functions	
Mapping on the ontology	
Mapping on a Slim Ontology	
Fitting in Depth	
Fitting in broad-based	
Keyword searching	
BLAST search	
To Map Against a 2nd List	
To Find corresponding IDs	

Table 7.5: Annotation options for both providing and managing functional annotation.

selection of either *Holm* or *Westfall–Young*.

Alternative methods are concerned with *Classification/Clustering* based on multivariate analyses, *Similarity/Distance Measures*, that allow measures for computing "differences" between GO categories, and *Other Methods*, such as Gene Set Enrichment Analysis, Time Series, etc.

Table 7.6 shows statistical methods and available options for the analysis based on the literature reviewed.

Analysis for	
A Single List	
A Query List vs a Reference List	
Multiple Lists	
Enrichment Analysis of GO Terms	
Under-represented	
Over-represented	
Define cutoff	
p-value	
q-value	
Test Used For The Analysis	
Binomial	Hypergeometric
Fisher's Exact Test	Permutation Test
Other	
Correction For Multiple Testing	
Bonferroni or Modified Bonferroni	
False Discovery Rate	Family Wise Error Rate
Benjamini–Hochberg	Holm
Benjamini–Yekutieli	Westfall–Young
Alternative Methods	
Classification/Clustering	
Similarity/Distance Measures	
Other Methods	

Table 7.6: Statistical methods and options provided by GO tools.

7.1.9 Output

After the analysis, GO tools provide the user with some results. There are different types of frameworks to show the resulting outputs. These outputs were organized into three main categories: *Format*, *Annotation Tables*, and *Visualization*.

Format features are intended to indicate what type of screen and/or file format is provided by the GO tool. The main file formats are: *HyperText Markup Language* (`.html`), *Spreadsheet program* (`.xls` or `.ods`), *Comma Separated Values* (`.csv`), *Semicolon Separated Values* (`.scsv`), *Tabulate*

Separated Values (`.tsv`), *Extensible Markup Language* (*.xml*), and *Other* formats

Annotation Tables refer to the characteristics of the table where results are being reported. These tables usually provide the *List of GO terms* and the associated *List of p_values*, which are *Limited* by either a *Number of Terms* or by *A p-value threshold.* Sometimes, *Annotations for clusters* of GO terms are also shown, and they can be *Hyperlinked with Cross–References.*

Visualization is intended to describe the method that has been used to display results. This group of capabilities shows a figure where the user can *View GO Terms* in a *Directed Acyclic Graph*, a *Tree* and/or *Bar Charts.* Sometimes this figure is interactive. That is, the GO tool has an associated *GO Browser*, that can be available in two forms: it is *Linked to the AmiGO browser* or the program has an *Own Browser Integrated.* GO programs also have the capability of showing the figure in such a way that the user can *View Genes Within Pathways*

Table 7.7 shows the characteristics of how GO tools report the results after the analysis.

Format		
HTML	Spreadsheet program	CSV
SCSV	TSV	XML
Other		
Annotation tables		
List of GO Terms	List of p-values	List limited by A Number of Terms A p-value Threshold
Annotations for clusters	Hyperlinked Cross–References	
Visualization		
View GO Terms Directed Acyclic Graph Tree Bar Charts	GO Browser Linked to the AmiGO browser Own browser integrated	View Genes Within Pathways

Table 7.7: Types of format files, features of annotation tables and characteristics of figures shown in outputs

7.2 Classification GO tools Based on the Standard Functionalities Set

26 GO tools were used and classified based on the Standard Functionalities Set. This classification is outlined in tables from 7.8 to 7.13, showing the following information:

- Table 7.8 classifies each GO tool based on the type of analysis that it can perform, the type of experiments associated with the information that can retrieve the tool, the type of the interface, the availability and the supported species.
- Table 7.9 reports the classification of the GO tools according to their input data.
- Table 7.10 shows the classification of GO tools according to the data sources or fields for retrieval annotation.
- Table 7.11 shows the the classification of the GO tools according to their functional annotations and possibilities managing this information.
- Tables 7.12 contains the methods that use each GO tool to perform enrichment analyses and/or alternatives their.
- Table 7.13 classifies each GO tool based on their outputs.

Cells in each table shows a symbol reporting whether a GO tool has a specific capability available. There are three types of symbols, which are: a point (•), indicating that a GO tool possesses the functionality, a question mark (?), denoting that a reference does at least exist where the capability is mentioned but it has not been possible to validate it *in situ*, and a blank ( ), meaning that the feature has not been found, neither in the references nor in the program.

	CLENCH	EASE	ermineJ	FatiGO	FuncAssociate	GARBAN	GFINDer	GOArray	GoMiner	GOstat	GoSurfer	GOTM	MAPPFinder	NetAffx	DAVID	MatchMiner	OntoGate (OntoBlast)	SOURCE	eGOn	GeneMerge	GOToolBox	Onto-Tools	Ontologizer	ontology Transverser	SeqExpress	THEA
TOOLS FOR																										
Exploration	●	●	●	●	●	●	●	●	●	●	●	●	●	●					●	●	●	●	●	●	●	●
Annotation															●	●	●	●	●	●	●	●	?	?	?	●
TYPE OF EXPERIMENT																										
DNA Microarrays	●	●	●	●	●	●	●		●	●	●	●	●	●	●	●		●	●	●	●	●		●	●	●
Spotted arrays: cDNA-Chip						●	?		●		?					●		●						●		
In situ arrays: GeneChip							●		●		●			●		●								●		
Proteomic		?	?	●		●			●		?	●	●		●	●						?				
SAGE experiment																						?			●	
CGH									?		?					?		?	?			?				
Others		●	?	●		?			●		●	●			?	●	●	●				?				
INTERFACE																										
Web-based tool			●	●	●	●	●			●		●		●	●	●	●	●	●	●	●	●		●		
Downloadable tool	●	●	●					●	●		●		●			●				●			●	●	●	●
Compatible Oss																										
Windows	●	●	●	●				●	●		●	●	●			●	●			●		●	●		●	●
Mac OS X			●					●	●							●				●		●	●			●
Unix			●					●	●							●				●		●	●			●
Linux	●		●	?				●	●							●				●		●	●			●
Command line (batch navigation)																●		●								
AVAILABILITY																										
Free	●			●			●		●		●	●	●	●		●	●				●			●	●	
Partially Free																						●				●
Fee																										
SAMPLE FILES/LISTS	●	●		●	●		●		?		●				●									●		
SUPPORTED SPECIES																										
Anopheles gambiae																	●									●
Arabidopsis thaliana	●			●	●					●											●	?				
Bacillus anthracis Ames																						?				
Bos taurus (Domestic Cow)																						●				
Caenorlabditis briggsae																										●
Caenorlabditis elegans				●	●												●				●	●				●
Coxiella burnetii RSA 493																						?				
Danio rerio (Zebrafish)												●										?				●
Dictyostellium discoideum																						?				
Drosophila melanogaster (Fly)				●	●						●	●			●		●				●	●				●
Fugu rubripes																										●
Geobacter sulfurreducens PCA																						?				
Glossina morsitans																						?				
Homo sapiens (Human)				●	●	●	●		●	●	●	●	●		●	●	●	●	●		●	●			●	●
Leishmania major																						?				
Mus musculus (Mouse)				●	●	●	●			●	●	●	●		●	?	●	●	●		●	●				●
Oryza sativa																						?				
Plasmodium falciparum																	●					?				
Pseudomonas syringae DC300																						?				
Rattus norvegicus (Rat)				●	●		●				●	●			●		●	●	●		●	●				●
Saccharomyces cervisiae (Beer)				●	●												●				●	?			●	
Schizosaccharomyces pombe (Yeast)					●					●			●				●					?				
Shewanella oneidensis					●																	?				
Trypanosoma brucei																						?				
Vibrio cholerae (Cholera)					●																	?				
Others				●						●																

Table 7.8: Classification of the GO tools according to Standard Functionalities Set for the Type of tool, Experiment, Availability and Supported Species, and Sample Files.

	CLENCH	EASE	ermineJ	FatiGO	FuncAssociate	GARBAN	GFINDer	GOArray	GoMiner	GOstat	GoSurfer	GOTM	MAPPFinder	NetAffx	DAVID	MatchMiner	OntoGate (OntoBlast)	SOURCE	eGOn	GeneMerge	GOToolBox	Onto-Tools	Ontologizer	ontology Transverser	SeqExpress	THEA
DATA																										
Automated Updating Source						●	●				●	●						●			●	●				
Weekly						●																				
Monthly									●	●											●		●			
Quarterly																●										
Non-defined																	●									
Inputs																										
A list of																										
Gene	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●		●	●	●	●	●	●	●	●	●
Proteins						●	●																			
Slim Terms	●																				●					
A single term																	●	●				●				
Choose evidence codes																			●		●					
Descriptions																	●			●		●				
Identifiers																										
Affymetrix probe set IDs				●			●			●	●	●		●	●	●						●		●		
CloneID																		●				●				
Chromosome locations																●										
Ensembl ID				●								●														
FISH clone ID																●										
FlyBase ID				●																						
GenBank accession number				●			●			●			●		●	●		●	●			●		●		
GenePept accession															●											
Gene Names																●						●				
Gene Symbol				●								●										●				
Gene Ontology Consortium ID																						●				
GI accession				●											●											
HUGO gene names									●							●										
IMAGE Clone																●			●							
LocusLink Ids							●				●	●		●	●			●	●			●			●	
PIR accesion															●											
Protein accession				●																						
PubMed															●							●				
RefSeq IDs				●			●								●	?										
Swiss-Prot Ids				●						●		●			●											
Symbol				●																						
Synonyms				●																						
TIGR_CMR				●																						
UniGene							●			●	●	●			●	●		●	●			●				
Cluster ID																		●								
IDs							●			●	●	●			●	●			●			●				
Names																		●								
Symbol																		●								
Additional data input															●	●						●				
Others										●					●											
Load inputs from																										
A file										●	●	●	●	●	●			●	●		●	●				
Paste into a text area															●		●	●			●	●				

Table 7.9: Classification of the GO tools according to the Standard Functionalities Set for the input Data.

	CLENCH	EASE	ermineJ	FatiGO	FuncAssociate	GARBAN	GFINDer	GOArray	GoMiner	GOstat	GoSurfer	GOTM	MAPPFinder	NetAffx	DAVID	MatchMiner	OntoGate (OntoBlast)	SOURCE	eGOn	GeneMerge	GOToolBox	Onto-Tools	Ontologizer	ontology Transverser	SeqExpress	THEA
ANNOTATION																										
Data source or fields for information Retrieval																										
Affymetrix descriptions				?			•			•	•	•		•	•	•										•
BioCarta									•																	
Chromosome locations																•		•								
dbEST						•												•				•				
EBI-EMBL							•																			•
Ensembl				•		•						•					•					•				•
FlyBase																	•					•				•
GDB Human Genome Data Base							•																			
GeneBank				•		•				•					•	•		•				•				•
GeneCards				?					•						•			•								
GeneMap99																		•								
Gene Name												•										•				
Gene Symbol												•				•										
Gene Ontology Annotations	•		•	•		•	•		•	•	•	•	•	•	•		•	•	•		•	•	•		•	•
GO-Mouse Genome Databases						•																				
Gene Reference Into Function (GRIF)												•														
HomoloGene												•														
InterPro				?		•																				
LocusLink ID				?		•	•		•		•	•			•	•		•	•			•			•	•
KEGG				?			•		•			•			•							•				
Map location												•														
Mouse Genome Database (MGD)																										•
Mouse Genome Informatics (MGI)							•			•																
NetAffx														•												
Organism												•														
OMIM				?			•		?			•			•	•		•								•
PDB																										•
PFAM				?			•								•											•
Phenotype												•														
PIR/IProt															•											•
PlasmoDB																	•									
PubMed links							•		•			•										•				
Rat Genome Database (RGD)							•										•									•
RefSeq				?					?						•	•		•				•				
RHdb																		•								
Saccharomyces Genome Database (SGD)																	•									•
TAIR	•			?																						
TIGR						•																				
TrEMBL						•			?								•									
UCSC Human Genome Build																•										
UniGene				•			•		?	•	•	•			•	•		•	•			•				
UniProt ID				•		•	•		?	•		•			•		•	•								•
UniSTS							•																			
WormBase																	•									•
Functional summaries				?								•			•			•								
Own database									•			•	•								•	•				•
Other				•		•			•	•					•			•				•				

Table 7.10: Classification of the GO tools according to the Standard Functionalities Set for the Annotation (part I).

	CLENCH	EASE	ermineJ	FatiGO	FuncAssociate	GARBAN	GFINDer	GOArray	GoMiner	GOstat	GoSurfer	GOTM	MAPPFinder	NetAffx	DAVID	MatchMiner	OntoGate (OntoBlast)	SOURCE	eGOn	GeneMerge	GOToolBox	Onto-Tools	Ontologizer	ontology Transverser	SeqExpress	THEA
ANNOTATION (cont.)																										
Functional Annotations																										
Biological pathways							●		●				●		●			●				●				
Desease							●								●											
Functional categories												●			●			●								
General annotations									●			●			●			●				●				
Gene Ontology	●			●			●		●	●	●	●	●	●	●		●	●			●	●		●		●
Molecular Functions	●			●			●		●	●	●	●	●	●	●		●	●			●	●		●	?	●
Biological Processes	●			●					●	?	●	●	●		●		●	●			●	●		●		●
Cellular Components	●			●					●	?	●	●	●		●		●	●			●	●		●		●
Literature															●			●								
Protein domains							●							?	●			●								
Protein interactions															●			●								
Possibilities																										
Append																				●						
Retrieval	●		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
Remove											●		●						●							
Clustering options	●					●	●			●		●		●							●	●			●	●
GO terms filtering functions	●			●		●	●		?	●	●	●	●		●						●	●				●
Mapping on the ontology	●			●	●		●		●		●	●	●				●		●	●	●	●				●
Mapping on a slim ontology	●																				●					
Fitting in-depth				●		●	●		?	●	●	●	●		●						●	●				
Fitting in broad-based															●											
Preprocess to obtain gene names																●	●	●								
Data transformation			●																			●			●	●
Assess bias							●															●				●
Reduce redundancy									●				●			●						●				●
Keyword searching				●		●	●					●				●	●	●			●	●			●	
BLAST search						●											●									
To find overlap with a second list																●										
To find corresponding IDs																●	●	●								
Application Program Interface (API)																						●				
Ordered-input mode					●				●												●					
File export/import			●						●		●	●	●	●		●		●	●			●			●	
Integration (with R)																									●	

Table 7.11: Classification of the GO tools according to the Standard Functionalities Set for the Annotation (part II).

	CLENCH	EASE	ermineJ	FatiGO	FuncAssociate	GARBAN	GFINDer	GOArray	GoMiner	GOstat	GoSurfer	GOTM	MAPPFinder	NetAffx	DAVID	MatchMiner	OntoGate (OntoBlast)	SOURCE	eGOn	GeneMerge	GOToolBox	Onto-Tools	Ontologizer	ontology Transverser	SeqExpress	THEA
STATISTICAL ANALYSIS																										
Analysis for																										
A single gene list	●	?	●	●	●			●			●	●			●	●			●		●					●
Interesting list vs. Reference list	●	?		●			?		●	●	●	●			?	●			●	?	●			●		●
Multiple lists						●								●	?				●				?			
Enrichment of GO terms	●	●	●	●	●	●	●	●	●	●	●	●	●	●	?				●	●	●	●	●	●	●	●
Under represented				●	●				●	●		●	●								●					●
Over represented	●	●		●	●		●	●	●	●		●	●							●	●					●
Define cutoff for			●						?		●		●				●					●		●		●
p_value			●								●		●													●
q_value											●															
Test in use																										
Binomial	●						●												●		●	●		●		●
Chi-squared	●									●	●			●								●				
Hypergeometric	●						●					●								●	●	●		●		●
Fisher's Exact				●	●		●		●	●									●		●	●		●		
McNemars																			●							
Permutation				●				●			●			?												
Correction for multiple tests	?			●	●				?	●	●			?						●	●	●		●		
Bonferroni or modified Bonferroni																				●	●					
False Discovery Rate Method				●	●					●											●			●		
Assuming independence (Benjamin and Hocberg)				●						●											●			●		
Dropping independence (Benjamin and Yekutieli)				●	●					●											●					
Family Wise Error Rate method				●						●											●					
Holm										●											●					
Westfall and Young				●																						
Others																					●			●		
Classification				●		●	?																			●
Similarity/distance measures	?					●	?								?		●		●		●					

Table 7.12: Classification of the GO tools according to the Standard Functionalities Set for the Statistical Analysis.

	CLENCH	EASE	ermineJ	FatiGO	FuncAssociate	GARBAN	GFINDer	GOArray	GoMiner	GOstat	GoSurfer	GOTM	MAPPFinder	NetAffx	DAVID	MatchMiner	OntoGate (OntoBlast)	SOURCE	eGOn	GeneMerge	GOToolBox	Onto-Tools	Ontologizer	ontology Transverser	SeqExpress	THEA
OUTPUTS																										
Format																										
HTML	•			•	•	•	•			•		•				•	•						•	•	•	
TSV																								•		
XML																							•	•		
XSL																								•		
Tabular text				•					•	•	•		?			•			•	•		•			•	
Spreadsheet program									•							•										
Semicolon-delimited file																						•				
Annotation tables																										
List of GO terms	•			•				•	•	•	•	•	•								•					•
List of p_values	•							•	•	•	•										•					
List limited													•													
By the number of terms										•			•													
By a p_value threshold										•																
Annotations for clusters	•			•		•	•			•	•	•		•	•							•			•	•
Hyperlinked cross-references	•			•			•	?	•		•	•		•	•	•	•			•	•	•		•		•
Visualization	•		•	•	•	•	•	•	•	•	•	•	•	•	•			•	•		•	•		•	•	•
View GO terms	•		•	•			•		•		•	•	•	•	•			•	•			•		•		•
Directed Acyclic Graph	•								•		•	•		•	•							•		•		•
Tree			•	•					•		?	•	•	?	•		•	•	•		•	•				
Bar charts				•			•					•										•				
View genes within pathways						•	•		?				•		•							•		•		
GO Browser	•			•	•				•	•		•	•	•			•					•			•	•
Linked to AmiGO Browser	•			•	•					•				•			•									
Own browser integrated									•			•										•			•	•

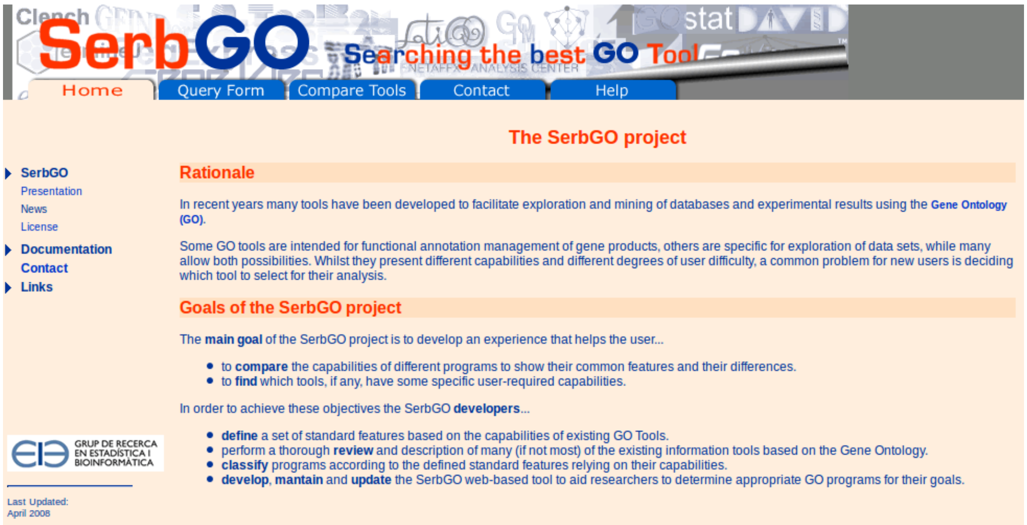
Table 7.13: Classification of the GO tools according to the Standard Functionalities Set for the Outputs.

7.3 A Web Tool for Selecting and Comparing GO Tools (SerbGO)

This section presents the tables of the database, the application outline of the web tool SerbGO, an example to illustrate the use and the benchmark of the resource

7.3.1 Availability of SerbGO

SerbGO is freely available under a Common Creative License (http://creativecommons.org/licenses/by/3.0/) and does not require a login. It can be accessed directly at the server (http://estbioinfo.stat.ub.es/apli/serbgo) of the *Statistics and Bioinformatics Research Group* (http://eib.stat.ub.edu/) led by Dr.Alex Sánchez. The tool was submitted and accepted to be available at the website of The GO Consortium (http://www.geneontology.org/GO.tools.microarray.shtml#serbgo). Figure 7.1 shows the main page of the SerbGO website.



The SerbGO project

Rationale

In recent years many tools have been developed to facilitate exploration and mining of databases and experimental results using the **Gene Ontology (GO)**.

Some GO tools are intended for functional annotation management of gene products, others are specific for exploration of data sets, while many allow both possibilities. Whilst they present different capabilities and different degrees of user difficulty, a common problem for new users is deciding which tool to select for their analysis.

Goals of the SerbGO project

The **main goal** of the SerbGO project is to develop an experience that helps the user...

- to **compare** the capabilities of different programs to show their common features and their differences.
- to **find** which tools, if any, have some specific user-required capabilities.

In order to achieve these objectives the SerbGO **developers**...

- **define** a set of standard features based on the capabilities of existing GO Tools.
- perform a thorough **review** and description of many (if not most) of the existing information tools based on the Gene Ontology.
- **classify** programs according to the defined standard features relying on their capabilities.
- **develop**, **mantain** and **update** the SerbGO web-based tool to aid researchers to determine appropriate GO programs for their goals.

Last Updated:
April 2008

Figure 7.1: SerbGO website.

7.3.2 The SerbGO Database

The database consists of a collection of seven tables with large lists of fields formally described and organized according to a relational model. These tables are:

- *main*: manages access to the other tables. It contains indexes, information about the name of the GO tools, the entities that are promoting the software, literature references that haven been reviewed, links to the resources and fields describing if tools are designed for exploration and/or annotation.
- *type*: fields describing the types of experiments and technologies to which each tool is intended for, the types of interfaces of the programs and which kind of availability is associated with each GO tool.
- *species*: fields associated with the list of species supported by each GO tool.
- *data*: consists of the information associated with the types of inputs required by each tool to do the corresponding analyses.
- *annotation*: this is the largest table, and stores information about the data sources from where each GO tool is fed, the types of functional annotations that are supported by each tool and the possibilities that they offer to manage such information.
- *statistics*: fields in this table are the list of statistical capabilities and methods associated with each GO tool.
- *outputs*: this is a table that contains information about the formats in which results are presented, as well as characteristics of the tables reported and also fields associated with the types of figures that are provided.

7.3.3 Application Workflow

7.3.3.1 Inputs

`SerbGO` offers two possibilities for execution: to search for and/or to compare GO tools. Both actions require to check some options in two different forms. The inputs required are:

1. To select a list of desired capacities from the Standard Functionalities Set in order to look for the GO tools that satisfy such capabilities.
2. To select a list of GO tools in order to yield a table for comparing such GO tools.

Both actions can be performed interactively using either the *Query Form* or the *Compare Tools* menu options

7.3.3.2 Query Form: Searching for GO tools

By clicking on the *Query Form* menu option, at the top of the page, the user accesses a form consisting of the Standard Functionalities Set arranged in nine sections (see section 7.1) and spread out over six pages. Each page allows the user to select required functionalities, and after validating the last page, to obtain a table listing the GO tools that satisfy the capabilities demanded. In detail, to find the "right tool", the user selects the desired functionalities by checking the appropriate fields in each specific section. Figures 7.2 to 7.7 show each of the form sections.

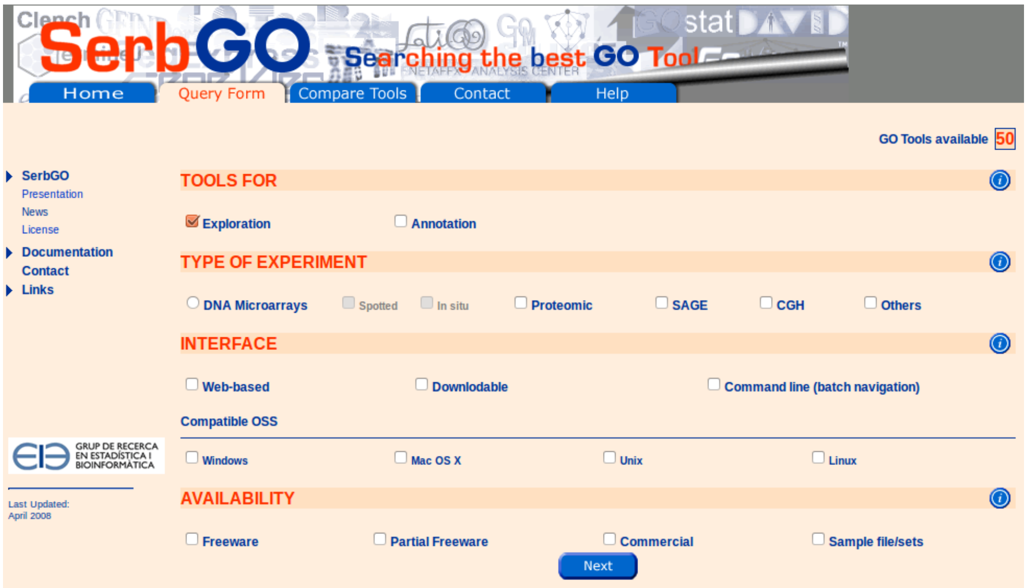
Once the choices have been made for a page it is necessary to validate the query by clicking on the *Next* button at the bottom of the page, which allows the user to move on to the following one. The next page will show the new sections, and the remaining number of tools available satisfying such features will appear at the top-right corner.

Note that in some sections, there are features shown as shaded colors. This means that such functionalities are non-available, unless they are activated. This can be done by switching on the corresponding previous radio button.

Queries are implemented with the logical operator `AND`. That is, the more capabilities there are selected, the fewer the tools that will be available.

During the process of navigation over the pages, and at any time, it is possible to start a new query if the user clicks on the Query Form menu option at the top of the page.

On the last selection page a *Find* button will appear instead of the *Next* button. This new button allows users to move on to the resulting outputs from the search after validation. Figure 7.7 shows the last page associated with the *Query Form*.



GO Tools available 50

TOOLS FOR

☑ Exploration □ Annotation

TYPE OF EXPERIMENT

○ DNA Microarrays □ Spotted □ In situ □ Proteomic □ SAGE □ CGH □ Others

INTERFACE

□ Web-based □ Downlodable □ Command line (batch navigation)

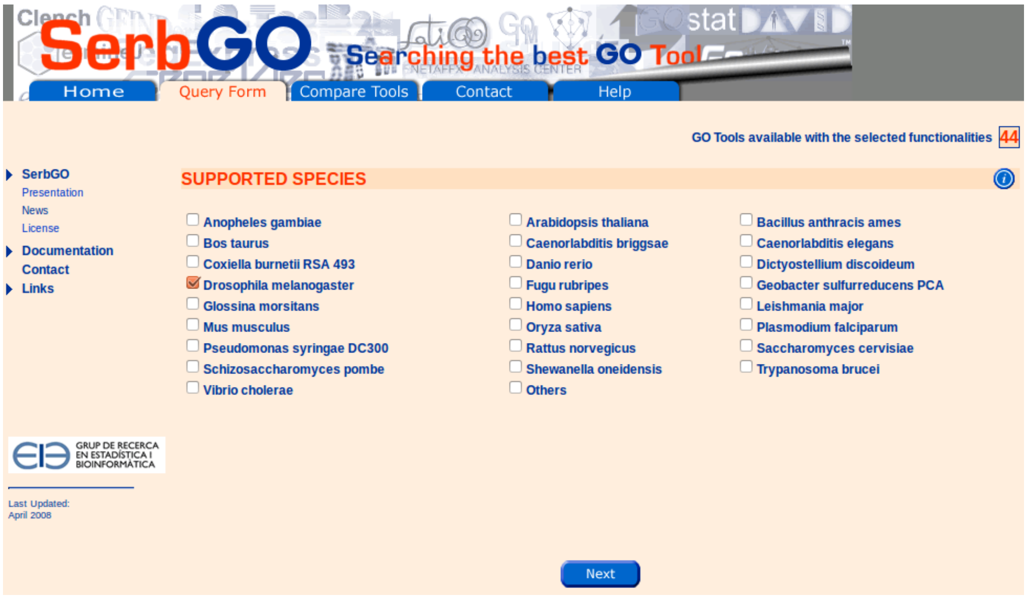
Compatible OSS

□ Windows □ Mac OS X □ Unix □ Linux

AVAILABILITY

□ Freeware □ Partial Freeware □ Commercial □ Sample file/sets

Figure 7.2: Screenshot of the *Query Form* showing the standard functionalities of sections TOOL FOR, TYPE OF EXPERIMENT, INTERFACE and AVAILABILITY.

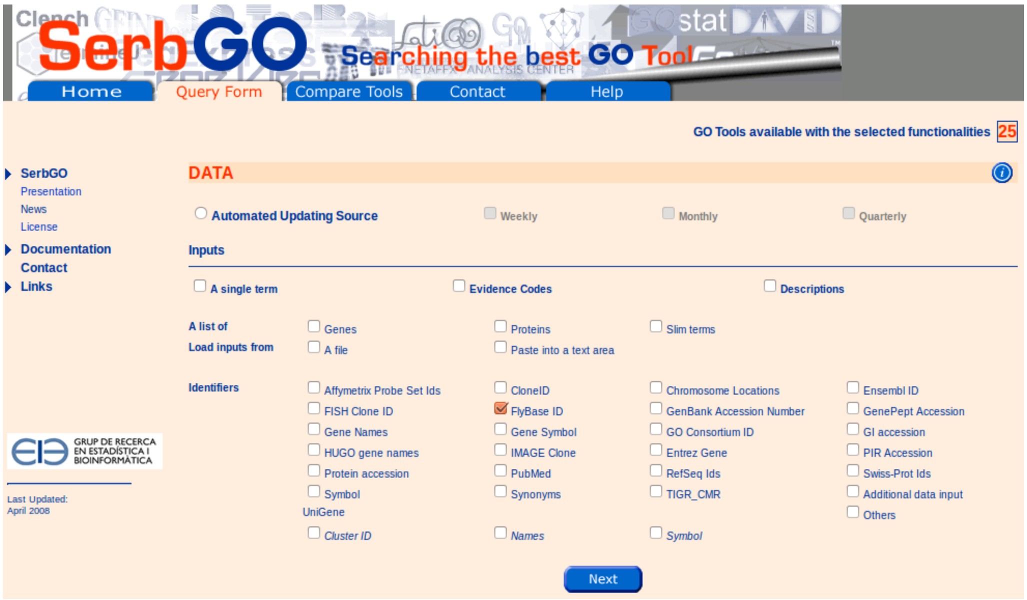


GO Tools available with the selected functionalities 44

SUPPORTED SPECIES

□ Anopheles gambiae
□ Bos taurus
□ Coxiella burnetii RSA 493
☑ Drosophila melanogaster
□ Glossina morsitans
□ Mus musculus
□ Pseudomonas syringae DC300
□ Schizosaccharomyces pombe
□ Vibrio cholerae
□ Arabidopsis thaliana
□ Caenorlabditis briggsae
□ Danio rerio
□ Fugu rubripes
□ Homo sapiens
□ Oryza sativa
□ Rattus norvegicus
□ Shewanella oneidensis
□ Others
□ Bacillus anthracis ames
□ Caenorlabditis elegans
□ Dictyostellium discoideum
□ Geobacter sulfurreducens PCA
□ Leishmania major
□ Plasmodium falciparum
□ Saccharomyces cervisiae
□ Trypanosoma brucei

Figure 7.3: Screenshot of the *Query Form* showing the standard functionalities of section SUPPORTED SPECIES.



GO Tools available with the selected functionalities 25

DATA

○ Automated Updating Source □ Weekly □ Monthly □ Quarterly

Inputs

□ A single term □ Evidence Codes □ Descriptions

A list of □ Genes □ Proteins □ Slim terms

Load inputs from □ A file □ Paste into a text area

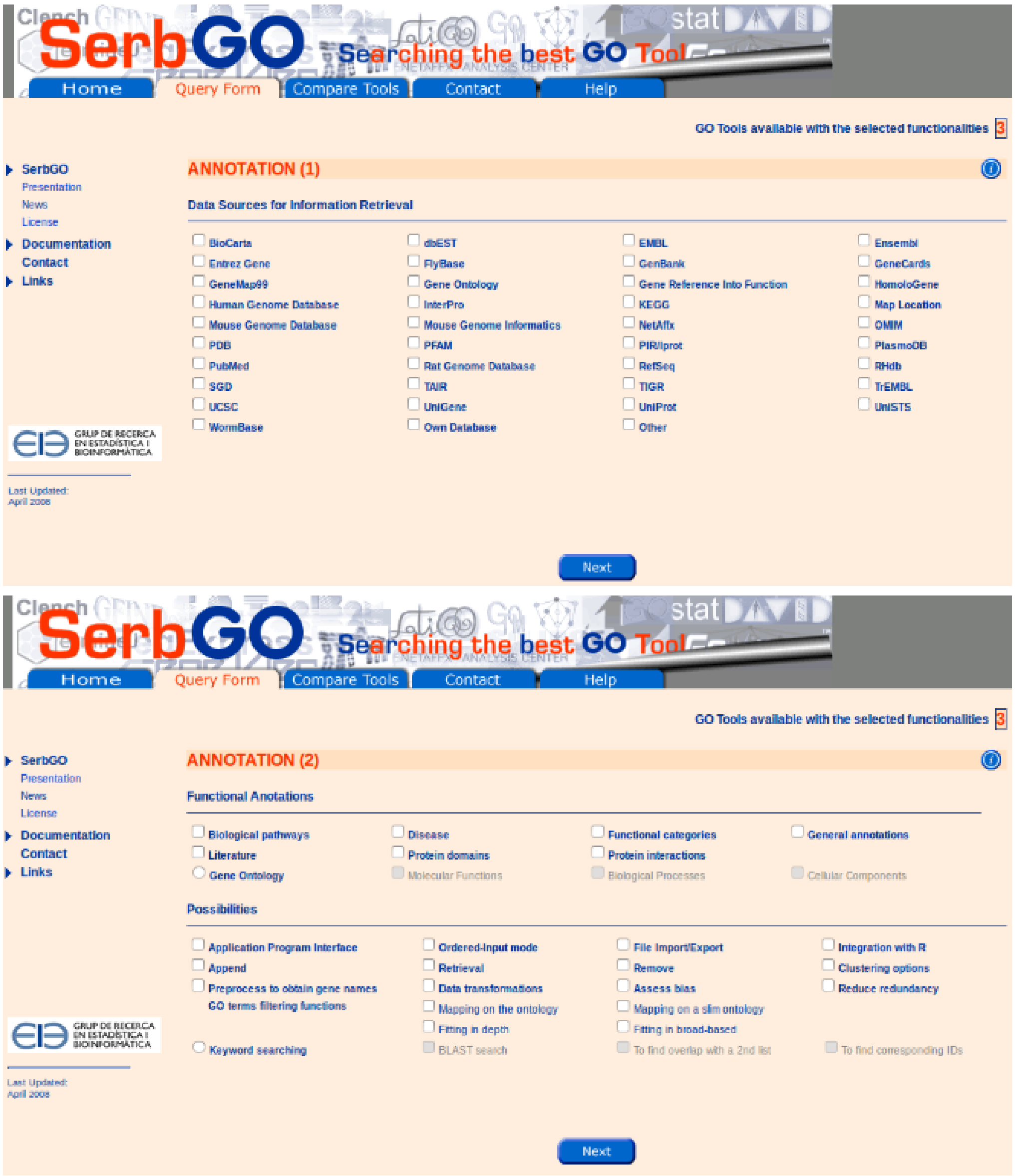
Identifiers

□ Affymetrix Probe Set Ids
□ FISH Clone ID
□ Gene Names
□ HUGO gene names
□ Protein accession
□ Symbol
□ CloneID
☑ FlyBase ID
□ Gene Symbol
□ IMAGE Clone
□ PubMed
□ Synonyms
□ Chromosome Locations
□ GenBank Accession Number
□ GO Consortium ID
□ Entrez Gene
□ RefSeq Ids
□ TIGR_CMR
□ Ensembl ID
□ GenePept Accession
□ GI accession
□ PIR Accession
□ Swiss-Prot Ids
□ Additional data input
□ Others

UniGene

□ *Cluster ID* □ *Names* □ *Symbol*

Figure 7.4: Screenshot of the *Query Form* showing the standard functionalities of section DATA.



GO Tools available with the selected functionalities 3

ANNOTATION (1)

Data Sources for Information Retrieval

□ BioCarta □ dbEST □ EMBL □ Ensembl
□ Entrez Gene □ FlyBase □ GenBank □ GeneCards
□ GeneMap99 □ Gene Ontology □ Gene Reference Into Function □ HomoloGene
□ Human Genome Database □ InterPro □ KEGG □ Map Location
□ Mouse Genome Database □ Mouse Genome Informatics □ NetAffx □ OMIM
□ PDB □ PFAM □ PIR/Iprot □ PlasmoDB
□ PubMed □ Rat Genome Database □ RefSeq □ RHdb
□ SGD □ TAIR □ TIGR □ TrEMBL
□ UCSC □ UniGene □ UniProt □ UniSTS
□ WormBase □ Own Database □ Other

Last Updated:
April 2008

GO Tools available with the selected functionalities 3

ANNOTATION (2)

Functional Anotations

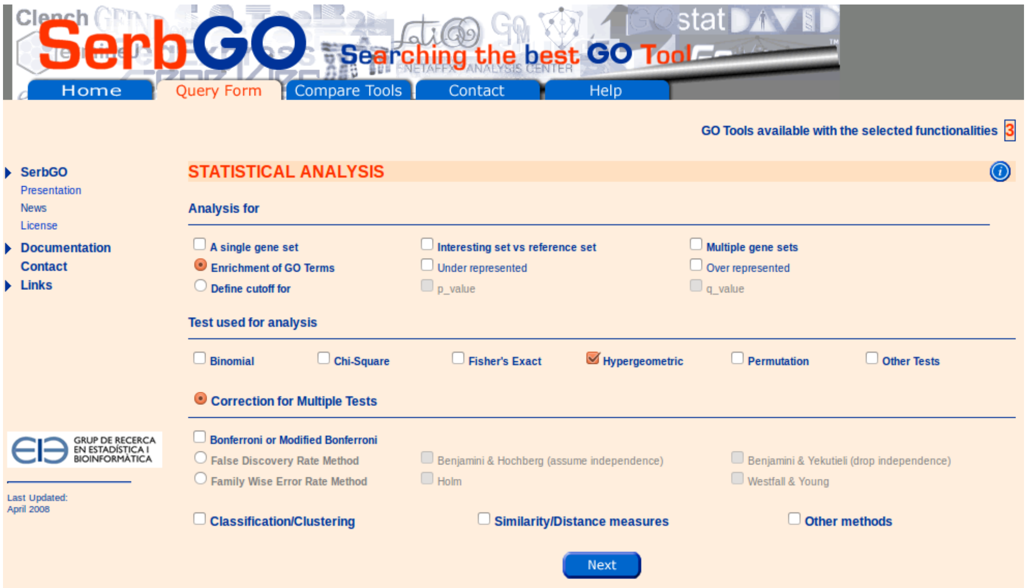
□ Biological pathways □ Disease □ Functional categories □ General annotations
□ Literature □ Protein domains □ Protein interactions
○ Gene Ontology □ Molecular Functions □ Biological Processes □ Cellular Components

Possibilities

□ Application Program Interface □ Ordered-Input mode □ File Import/Export □ Integration with R
□ Append □ Retrieval □ Remove □ Clustering options
□ Preprocess to obtain gene names □ Data transformations □ Assess bias □ Reduce redundancy
GO terms filtering functions □ Mapping on the ontology □ Mapping on a slim ontology
□ Fitting in depth □ Fitting in broad-based
○ Keyword searching □ BLAST search □ To find overlap with a 2nd list □ To find corresponding IDs

Last Updated:
April 2008

Figure 7.5: Screenshots of the *Query Form* showing the standard functionalities of sections ANNOTATION (1) and ANNOTATION (2).



GO Tools available with the selected functionalities 3

STATISTICAL ANALYSIS

Analysis for

□ A single gene set □ Interesting set vs reference set □ Multiple gene sets
● Enrichment of GO Terms □ Under represented □ Over represented
○ Define cutoff for □ p_value □ q_value

Test used for analysis

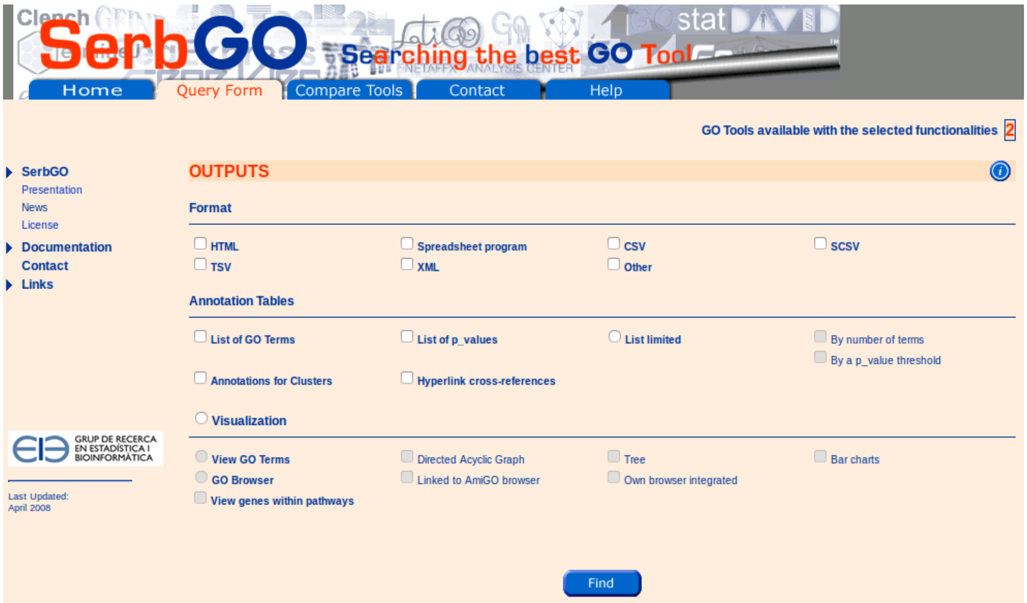
□ Binomial □ Chi-Square □ Fisher's Exact ☑ Hypergeometric □ Permutation □ Other Tests

● Correction for Multiple Tests

□ Bonferroni or Modified Bonferroni
○ False Discovery Rate Method □ Benjamini & Hochberg (assume independence) □ Benjamini & Yekutieli (drop independence)
○ Family Wise Error Rate Method □ Holm □ Westfall & Young

□ Classification/Clustering □ Similarity/Distance measures □ Other methods

Figure 7.6: Screenshot of the *Query Form* showing the standard functionalities of section STATISTICAL ANALYSIS.



GO Tools available with the selected functionalities 2

OUTPUTS

Format

□ HTML □ Spreadsheet program □ CSV □ SCSV
□ TSV □ XML □ Other

Annotation Tables

□ List of GO Terms □ List of p_values ○ List limited □ By number of terms
□ By a p_value threshold
□ Annotations for Clusters □ Hyperlink cross-references

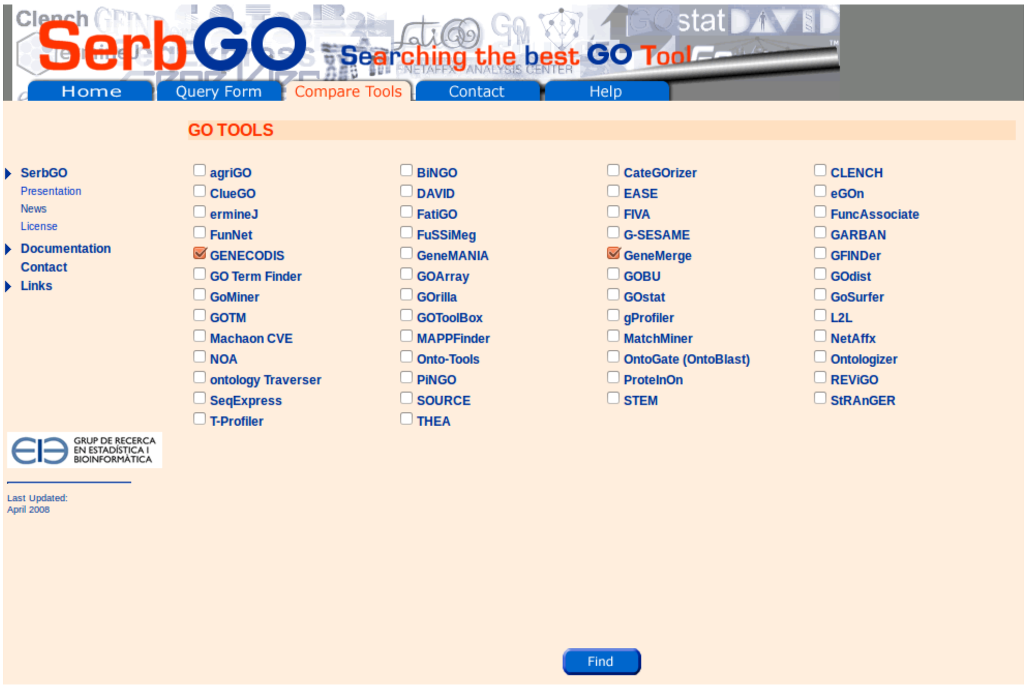
○ Visualization

○ View GO Terms □ Directed Acyclic Graph □ Tree □ Bar charts
○ GO Browser □ Linked to AmiGO browser □ Own browser integrated
□ View genes within pathways

Figure 7.7: Screenshot of the *Query Form* showing the standard functionalities of section OUTPUTS.

7.3.3.3 Compare Tools Form: comparing classified GO tools

By checking any of the tools in the *Compare Tools* form, a list of their capabilities according to the Standard Functionalities Set can be obtained. Figure 7.8 shows an screenshot of this form, where the alphabetically listed GO Tools are the programs available at the end of this thesis.



GO TOOLS

- □ agriGO
- □ ClueGO
- □ ermineJ
- □ FunNet
- ☑ GENECODIS
- □ GO Term Finder
- □ GoMiner
- □ GOTM
- □ Machaon CVE
- □ NOA
- □ ontology Traverser
- □ SeqExpress
- □ T-Profiler
- □ BiNGO
- □ DAVID
- □ FatiGO
- □ FuSSiMeg
- □ GeneMANIA
- □ GOArray
- □ GOrilla
- □ GOToolBox
- □ MAPPFinder
- □ Onto-Tools
- □ PiNGO
- □ SOURCE
- □ THEA
- □ CateGOrizer
- □ EASE
- □ FIVA
- □ G-SESAME
- ☑ GeneMerge
- □ GOBU
- □ GOstat
- □ gProfiler
- □ MatchMiner
- □ OntoGate (OntoBlast)
- □ ProteInOn
- □ STEM
- □ CLENCH
- □ eGOn
- □ FuncAssociate
- □ GARBAN
- □ GFINDer
- □ GOdist
- □ GoSurfer
- □ L2L
- □ NetAffx
- □ Ontologizer
- □ REViGO
- □ StRAnGER

Last Updated:
April 2008

Figure 7.8: Screenshot of the *Compare Tools* form showing the list of GO Tools stored in the SerbGO database.

7.3.3.4 Outputs

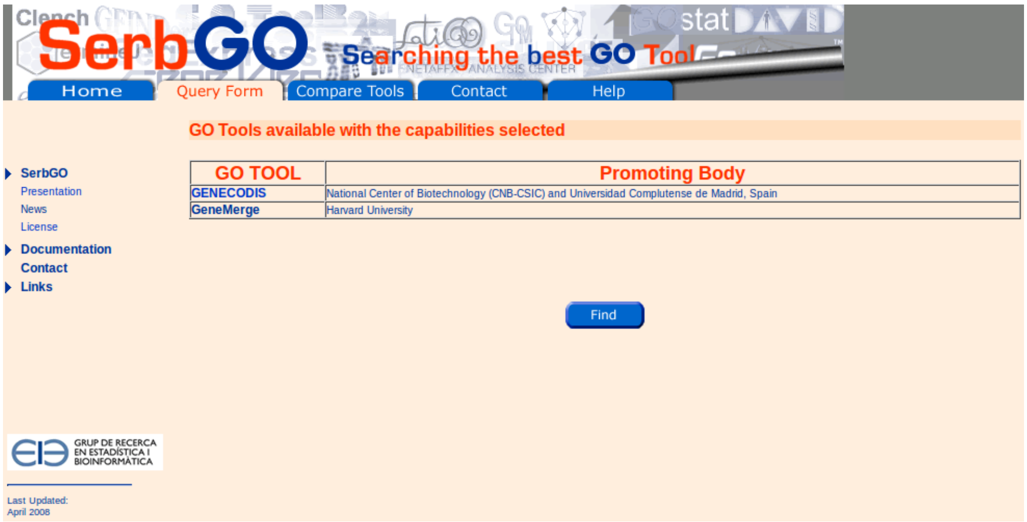
The output of the *Query Form* is a table with two columns (see figure 7.9). The first column shows the list of GO Tool names sorted alphabetically, and the second column shows the Promoter Body of the GO tool. Each GO tool name is linked to the corresponding website. Thus by clicking on the name, a new tab will be opened in the web browser with the tool already opened. The list of programs shown in that table may be compared by clicking the *Find* button placed at the bottom of the page. This action leads the user to a new page showing a table such that the rows are the functionalities and the columns are the GO Tool names, which are also linked to their respective sites (see figure 7.10).

The output page of the *Compare Tools* form shows a table like the comparison stated above (see figure 7.10), but instead of comparing the GO tools provided by the SerbGO search robot, the GO tools have been selected in the *Compare Tools* form.

7.3.3.5 Example

To illustrate the concept of how to determine which GO tools for gene enrichment analysis provide the features required by a potential user, the following example is considered. A potential SerbGO user has a list of *Drosophila melanogaster* genes. The user would like to know which tools are available that could perform a GO enrichment analysis for a list of FlyBase IDs, based on the hypergeometric distribution test and p-values that are corrected for multiple testing. In such a situation, the user should click on the *Query Form* menu option and select “Exploration” in the TOOLS FOR section (figure 7.2). Then, move on the next page and select the “Drosophila melanogaster” option (figure 7.3). After validation, there are 25 potential tools available. In the DATA section, the user checks “FlyBase ID” identifiers (figure 7.4), and he/she has to continue until the STATISTICAL ANALYSIS section, where he/she will select “Enrichment of GO Terms”, “Hypergeometric” test and “Correction for Multiple Tests” (figure 7.7). When the user moves on to the last query page, he/she clicks on the *Find* button and the outputs page shows the resulting table. Two GO tools with the capabilities required by the user are classified in the SerbGO database. These tools are GENECODIS ([24]) and GeneMerge ([26]) (figure 7.9).

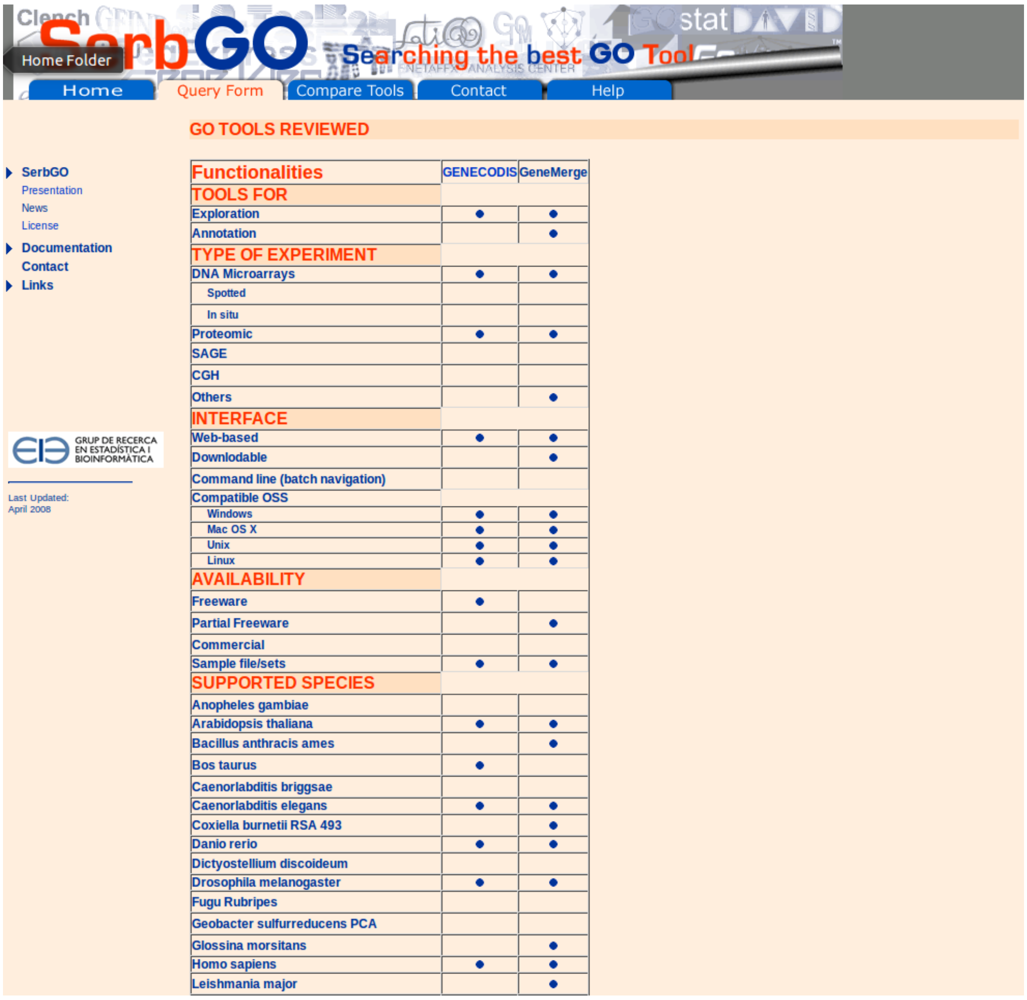
If the user wishes to compare the resulting tools, it can be done in two ways. First, just by simply clicking on the *Find* button at the bottom of the page, and second by selecting both tools on the *Compare Tools* form (figure 7.8). In both cases, the resulting action yields a table where rows are the standard functionalities, columns are the names of the GO tools. That is, in this example GENECODIS and GeneMerge are linked to the respective sites. In each cell from the resulting table a dot is shown when the capability of a tool is available, otherwise the cell is empty (see figure 7.10).



GO Tools available with the capabilities selected

GO TOOL	Promoting Body
GENECODIS	National Center of Biotechnology (CNB-CSIC) and Universidad Complutense de Madrid, Spain
GeneMerge	Harvard University

Figure 7.9: Screenshot of the GO Tools classified that satisfy the requirements of a user.



GO TOOLS REVIEWED

Functionalities	GENECODIS	GeneMerge
TOOLS FOR		
Exploration	●	●
Annotation		●
TYPE OF EXPERIMENT		
DNA Microarrays	●	●
Spotted		
In situ		
Proteomic	●	●
SAGE		
CGH		
Others		●
INTERFACE		
Web-based	●	●
Downlodable		●
Command line (batch navigation)		
Compatible OSS		
Windows	●	●
Mac OS X	●	●
Unix	●	●
Linux	●	●
AVAILABILITY		
Freeware	●	
Partial Freeware		●
Commercial		
Sample file/sets	●	●
SUPPORTED SPECIES		
Anopheles gambiae		
Arabidopsis thaliana	●	●
Bacillus anthracis ames		●
Bos taurus	●	
Caenorlabditis briggsae		
Caenorlabditis elegans	●	●
Coxiella burnetii RSA 493		●
Danio rerio	●	●
Dictyostellium discoideum		
Drosophila melanogaster	●	●
Fugu Rubripes		
Geobacter sulfurreducens PCA		
Glossina morsitans		●
Homo sapiens	●	●
Leishmania major		●

Figure 7.10: Screenshot of the available capabilities for two GO Tools.

7.3.4 Benchmark

During the testing period, most of the tools available at The GO Consortium website were included in the beta version. This process included 26 GO Tools listed in table 6.1 and classified in tables from 7.8 to 7.13. This beta version was used by several people from different organizations around the world. SerbGO was also tested by developers of some of these tools, such as FatiGO ([3]), GARBAN ([105]), o BiNGO ([99]), who suggested some improvements that were incorporated into the testing version and validated in the first stable version.

SerbGO has been running since June 2006 and was published in 2008 ([111]). It has been updated, but not periodically, for technical reasons. At the end of this thesis the serbGO database stored information on about 50 GO tools.

7.4 Evolution and Clustering of GO Tools

The following subsections present the results of the statistical analysis, aimed at understanding how the capabilities of GO tools, based on the classification of the Standard Functionalities Set 6.2, have been evolving and regrouping (if so) in clusters over time.

7.4.1 Descriptive Statistics

Descriptive statistics analysis has been divided in two a *Global Analysis* and an *Analysis by Sections* of functionalities. The following paragraph presents the results obtained in each part of the analysis.

Global Analysis

After homogenizing the tables downloaded from SerbGO database, and eliminating “redundancies” from the original 205 standard functionalities 6.4.1.2, 26 GO tools are classified according to 178 independent functionalities. Table 7.14 summarizes the capabilities of each GO tool per year showing absolute and relative frequencies. Briefly, in 2005, Onto-Tools ([46], [88]) was the tool with most capabilities, and EASE ([40]) was the tool with least functionalities available. In 2007, most of the GO tools introduced new features, and Onto-Tools was again the software with most functionalities available. However, GOArray did not actually introduce new characteristics or improve the old ones. In 2009, most of the tools once again improved their capabilities.

The GO tool with most functionalities was DAVID ([40]), covering about 61% of the Standard Functionalities Set analyzed.

GO Tool	2005		2007		2099	
	n	%	n	%	n	%
CLENCH	29	16.292	44	24.719	48	26.966
EASE	8	4.494	65	36.517	75	42.135
ermineJ	17	9.551	56	31.461	61	34.27
FatiGO	53	29.775	60	33.708	86	48.315
FuncAssociate	24	13.483	40	22.472	44	24.719
GARBAN	31	17.416	66	37.079	66	37.079
GFINDer	48	26.966	90	50.562	100	56.18
GOArray	14	7.865	14	7.865	14	7.865
GoMiner	39	21.91	86	48.315	94	52.809
GOstat	36	20.225	51	28.652	77	43.258
GoSurfer	40	22.472	54	30.337	56	31.461
GOTM	57	32.022	58	32.584	63	35.393
MAPPFinder	27	15.169	31	17.416	78	43.82
NetAffx	21	11.798	81	45.506	70	39.326
DAVID	53	29.775	77	43.258	109	61.236
MatchMiner	44	24.719	56	31.461	61	34.27
OntoGate (OntoBlast)	37	20.787	38	21.348	41	23.034
SOURCE	43	24.157	48	26.966	56	31.461
eGOn	31	17.416	55	30.899	82	46.067
GeneMerge	19	10.674	50	28.09	71	39.888
GOToolBox	40	22.472	49	27.528	83	46.629
Onto-Tools	69	38.764	93	52.247	103	57.865
Ontologizer	13	7.303	58	32.584	65	36.517
ontology Traverser	24	13.483	27	15.169	31	17.416
SeqExpress	23	12.921	46	25.843	75	42.135
THEA	54	30.337	64	35.955	77	43.258

Table 7.14: Absolute and relative frequencies of available functionalities in the GO tools per year

Figure 7.11 shows a bar plot of the percentages of functionalities that are available for each GO tool and year. There are some differences between years, in terms of distributions of frequencies. Note that there is a considerable increase in functionalities between the years 2005 and 2007, and even though there is also an increase in percentages between 2007 and 2009, the change is not as remarkable as the previous one.

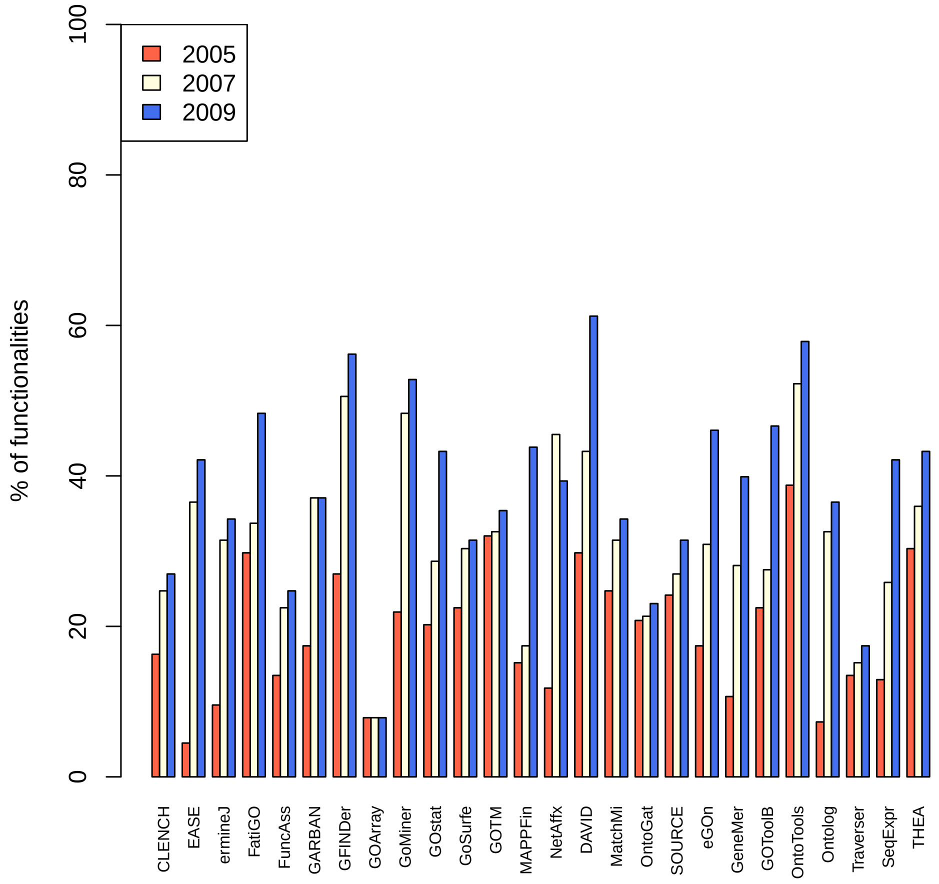


Figure 7.11: Bar plot of the percentage of available functionalities in the GO Tools per year.

Analysis by Sections

For this analysis, functionalities have been classified into six sections. Table 7.15 shows the sections and number of functionalities after the homogenization included in each one.

Tables 7.16, 7.17 and 7.17 show absolute and relative frequencies of capabilities per GO tool and year, separated by sections of functionalities. These

Section	Num. of Functionalities
Type of Tool	18
Supported Specie	26
Input Data	35
Annotation	72
Statistical analysis	14
Output	13
TOTAL	178

Table 7.15: Number of standard functionalities per section.

tables are supported with bar diagrams (one per section), like in the global analysis 7.4.1, displaying the distributions of the frequencies for each year. All these plots are shown in figure 7.12.

A bird's eye view of bar plots suggest that all the sections of functionalities have experienced relevant changes. It seems that *Annotation Functionalities* and *Supported Species* are the sections where promoters have invested most effort. Note that in the associated bar plots, bars in blue (i.e. percentages of functionalities in 2009) are higher than the red ones (e.g. percentages of functionalities in 2005). That is, the percentage of functionalities available in 2009 for these sections is considerably higher than the percentage of functionalities in 2005. However, with bar plots associated with *Type of Tool*, *Statistical Analysis*, and *Outputs*, even though the percentages of GO tool functionalities increased, such changes seem to be more "homogeneous". The bar plot of *Input Data* functionalities does not suggest great changes of functionalities for GO tools *a priori*.

Section	GO Tool	2005		2007		2009	
		n	%	n	%	n	%
Type of Tool	CLENCH	7	38.889	7	38.889	7	38.889
	EASE	6	33.333	7	38.889	7	38.889
	ermineJ	8	44.444	14	77.778	14	77.778
	FatiGO	8	44.444	6	33.333	10	55.556
	FuncAssociate	4	22.222	5	27.778	5	27.778
	GARBAN	4	22.222	7	38.889	7	38.889
	GFINDer	5	27.778	7	38.889	11	61.111
	GOArray	6	33.333	6	33.333	6	33.333
	GoMiner	10	55.556	12	66.667	13	72.222
	GOstat	3	16.667	4	22.222	8	44.444
	GoSurfer	7	38.889	8	44.444	8	44.444
	GOTM	7	38.889	6	33.333	10	55.556
	MAPPFinder	6	33.333	6	33.333	9	50.000
	NetAffx	4	22.222	7	38.889	10	55.556
	DAVID	5	27.778	7	38.889	13	72.222
	MatchMiner	12	66.667	13	72.222	12	66.667
	OntoGate (OntoBlast)	5	27.778	4	22.222	8	44.444
	SOURCE	5	27.778	6	33.333	10	55.556
	eGOn	4	22.222	8	44.444	12	66.667
	GeneMerge	9	50.000	13	72.222	13	72.222
	GOToolBox	5	27.778	7	38.889	12	66.667
	Onto-Tools	9	50.000	13	72.222	13	72.222
	Ontologizer	6	33.333	12	66.667	12	66.667
	ontology Traverser	6	33.333	5	27.778	5	27.778
	SeqExpress	6	33.333	9	50.000	9	50.000
	THEA	9	50.000	10	55.556	10	55.556
Supported Specie	CLENCH	1	3.846	1	3.846	1	3.846
	EASE	0	0	8	30.769	8	30.769
	ermineJ	0	0	3	11.538	3	11.538
	FatiGO	8	30.769	8	30.769	11	42.308
	FuncAssociate	10	38.462	11	42.308	13	50.000
	GARBAN	2	7.692	2	7.692	2	7.692
	GFINDer	3	11.538	13	50.000	13	50.000
	GOArray	0	0	0	0	0	0
	GoMiner	1	3.846	13	50.000	13	50.000
	GOstat	5	19.231	7	26.923	18	69.231
	GoSurfer	4	15.385	5	19.231	5	19.231
	GOTM	5	19.231	5	19.231	6	23.077
	MAPPFinder	3	11.538	3	11.538	11	42.308
	NetAffx	0	0	9	34.615	11	42.308
	DAVID	4	15.385	4	15.385	8	30.769
	MatchMiner	1	3.846	2	7.692	2	7.692
	OntoGate (OntoBlast)	9	34.615	9	34.615	6	23.077
	SOURCE	3	11.538	3	11.538	3	11.538
	eGOn	3	11.538	12	46.154	13	50.000
	GeneMerge	0	0	19	73.077	20	76.923
	GOToolBox	7	26.923	7	26.923	19	73.077
	Onto-Tools	6	23.077	20	76.923	20	76.923
	Ontologizer	0	0	23	88.462	23	88.462
	ontology Traverser	0	0	3	11.538	3	11.538
	SeqExpress	2	7.692	4	15.385	4	15.385
	THEA	9	34.615	10	38.462	10	38.462

Table 7.16: Frequencies of functionalities of the GO tools per year by sections *Type of Tool* and *Supported Specie*.

Section	GO Tool	2005		2007		2009	
		n	%	n	%	n	%
Input Data	CLENCH	2	5.714	8	22.857	8	22.857
	EASE	1	2.857	7	20.000	14	40.000
	ermineJ	1	2.857	5	14.286	5	14.286
	FatiGO	13	37.143	16	45.714	21	60.000
	FuncAssociate	1	2.857	9	25.714	10	28.571
	GARBAN	3	8.571	9	25.714	8	22.857
	GFINDer	8	22.857	13	37.143	15	42.857
	GOArray	1	2.857	1	2.857	1	2.857
	GoMiner	2	5.714	10	28.571	13	37.143
	GOstat	8	22.857	10	28.571	11	31.429
	GoSurfer	6	17.143	6	17.143	6	17.143
	GOTM	9	25.714	11	31.429	10	28.571
	MAPPFinder	3	8.571	3	8.571	14	40.000
	NetAffx	4	11.429	12	34.286	14	40.000
	DAVID	15	42.857	14	40.000	20	57.143
	MatchMiner	11	31.429	19	54.286	23	65.714
	OntoGate (OntoBlast)	4	11.429	6	17.143	6	17.143
	SOURCE	9	25.714	9	25.714	10	28.571
	eGOn	7	20.000	10	28.571	10	28.571
	GeneMerge	2	5.714	7	20.000	11	31.429
	GOToolBox	6	17.143	8	22.857	12	34.286
	Onto-Tools	16	45.714	16	45.714	14	40.000
	Ontologizer	2	5.714	7	20.000	7	20.000
	ontology Traverser	3	8.571	4	11.429	4	11.429
	SeqExpress	2	5.714	11	31.429	12	34.286
	THEA	1	2.857	5	14.286	5	14.286
Annotation	CLENCH	7	9.722	12	16.667	15	20.833
	EASE	0	0.000	30	41.667	33	45.833
	ermineJ	4	5.556	18	25.000	21	29.167
	FatiGO	11	15.278	17	23.611	29	40.278
	FuncAssociate	3	4.167	6	8.333	6	8.333
	GARBAN	15	20.833	34	47.222	35	48.611
	GFINDer	24	33.333	38	52.778	42	58.333
	GOArray	1	1.389	1	1.389	1	1.389
	GoMiner	17	23.611	37	51.389	38	52.778
	GOstat	11	15.278	16	22.222	25	34.722
	GoSurfer	10	13.889	20	27.778	21	29.167
	GOTM	27	37.500	27	37.500	27	37.500
	MAPPFinder	10	13.889	13	18.056	34	47.222
	NetAffx	7	9.722	41	56.944	33	45.833
	DAVID	25	34.722	36	50.000	51	70.833
	MatchMiner	14	19.444	18	25.000	20	27.778
	OntoGate (OntoBlast)	14	19.444	14	19.444	16	22.222
	SOURCE	25	34.722	25	34.722	28	38.889
	eGOn	7	9.722	7	9.722	29	40.278
	GeneMerge	3	4.167	3	4.167	19	26.389
	GOToolBox	10	13.889	10	13.889	23	31.944
	Onto-Tools	26	36.111	26	36.111	38	52.778
	Ontologizer	2	2.778	2	2.778	9	12.500
	ontology Traverser	2	2.778	2	2.778	6	8.333
	SeqExpress	8	11.111	8	11.111	35	48.611
	THEA	24	33.333	24	33.333	37	51.389

Table 7.17: Frequencies of functionalities of the GO tools per year by sections *Input Data* and *Annotation*.

Section	GO Tool	2005		2007		2009	
		n	%	n	%	n	%
Statistical Analysis	CLENCH	6	42.857	10	71.429	10	71.429
	EASE	1	7.143	6	42.857	6	42.857
	ermineJ	3	21.429	8	57.143	9	64.286
	FatiGO	7	50.000	6	42.857	7	50.000
	FuncAssociate	4	28.571	5	35.714	6	42.857
	GARBAN	4	28.571	7	50.000	7	50.000
	GFINDer	4	28.571	11	78.571	11	78.571
	GOArray	3	21.429	3	21.429	3	21.429
	GoMiner	3	21.429	6	42.857	7	50.000
	GOstat	5	35.714	7	50.000	8	57.143
	GoSurfer	7	50.000	8	57.143	8	57.143
	GOTM	4	28.571	4	28.571	4	28.571
	MAPPFinder	2	14.286	2	14.286	6	42.857
	NetAffx	3	21.429	5	35.714	0	0
	DAVID	1	7.143	8	57.143	9	64.286
	MatchMiner	2	14.286	0	0	0	0.000
	OntoGate (OntoBlast)	2	14.286	2	14.286	2	14.286
	SOURCE	0	0	0	0	0	0
	eGOn	8	57.143	9	64.286	9	64.286
	GeneMerge	3	21.429	4	28.571	4	28.571
	GOToolBox	8	57.143	10	71.429	10	71.429
	Onto-Tools	7	50.000	10	71.429	10	71.429
	Ontologizer	1	7.143	6	42.857	6	42.857
	ontology Traverser	7	50.000	7	50.000	7	50.000
	SeqExpress	1	7.143	7	50.000	7	50.000
	THEA	7	50.000	8	57.143	8	57.143
Output	CLENCH	6	46.154	6	46.154	7	53.846
	EASE	0	0.000	7	53.846	7	53.846
	ermineJ	1	7.692	8	61.538	9	69.231
	FatiGO	6	46.154	7	53.846	8	61.538
	FuncAssociate	2	15.385	4	30.769	4	30.769
	GARBAN	3	23.077	7	53.846	7	53.846
	GFINDer	4	30.769	8	61.538	8	61.538
	GOArray	3	23.077	3	23.077	3	23.077
	GoMiner	6	46.154	8	61.538	10	76.923
	GOstat	4	30.769	7	53.846	7	53.846
	GoSurfer	6	46.154	7	53.846	8	61.538
	GOTM	5	38.462	5	38.462	6	46.154
	MAPPFinder	3	23.077	4	30.769	4	30.769
	NetAffx	3	23.077	7	53.846	2	15.385
	DAVID	3	23.077	8	61.538	8	61.538
	MatchMiner	4	30.769	4	30.769	4	30.769
	OntoGate (OntoBlast)	3	23.077	3	23.077	3	23.077
	SOURCE	1	7.692	5	38.462	5	38.462
	eGOn	2	15.385	9	69.231	9	69.231
	GeneMerge	2	15.385	4	30.769	4	30.769
	GOToolBox	4	30.769	7	53.846	7	53.846
	Onto-Tools	5	38.462	8	61.538	8	61.538
	Ontologizer	2	15.385	8	61.538	8	61.538
	ontology Traverser	6	46.154	6	46.154	6	46.154
	SeqExpress	4	30.769	7	53.846	8	61.538
	THEA	4	30.769	7	53.846	7	53.846

Table 7.18: Frequencies of functionalities of GO tools per year for sections *Statistical Analysis* and *Output*.

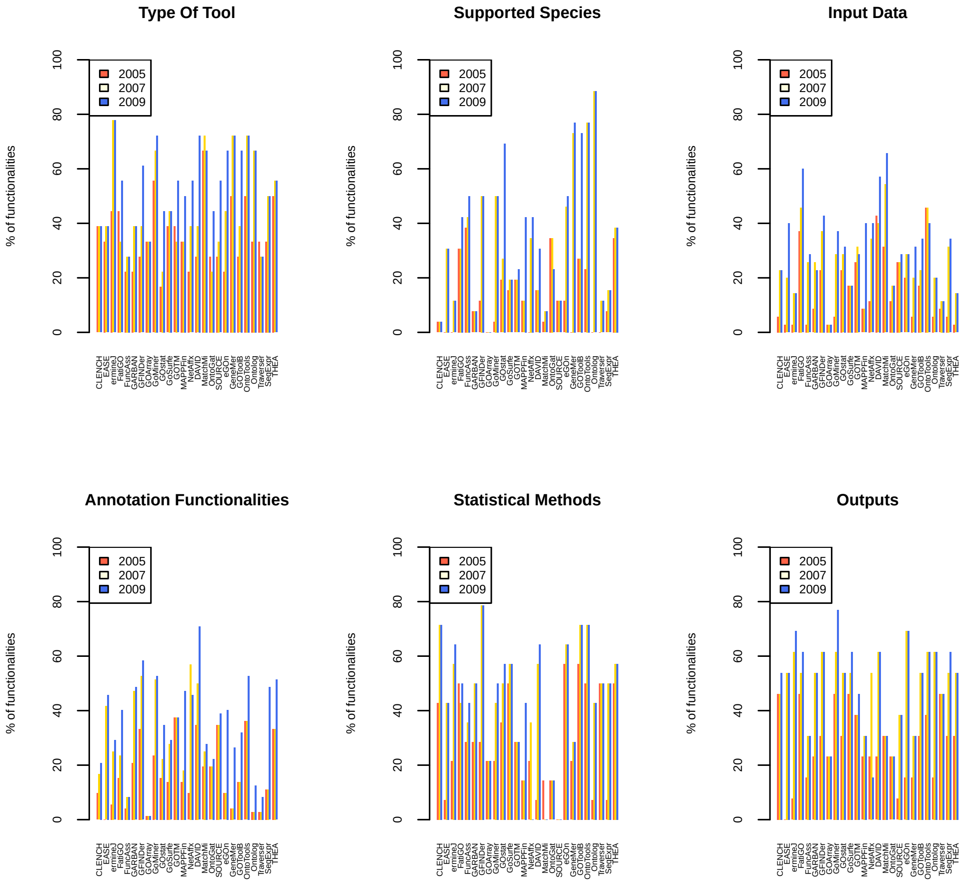


Figure 7.12: Bar plots of the percentage of functionalities of GO Tools per year by sections.

7.4.2 Inferential Analysis

Global Analysis

Table 7.19 shows the results of the Chi-squared tests of homogeneity. The comparisons between years 2005 *vs.* 2007 and 2005 *vs.* 2009 are statistically significant because, in both cases, adjusted p-values (i.e. 5.763450e-10 and 1.750368e-10 respectively) are less than a level of significance of $\alpha = 0.05$. Therefore, in both cases, we reject the null hypothesis that the distribution of frequencies between each pair of years is the same. However, there is not enough evidence to reject the null hypothesis associated with the comparison 2007 *vs.* 2009, because the adjusted p-value (i.e. 1.106518e-01) is greater than a level of significance of $\alpha = 0.05$.

Comparison	Chi.Squared	df	PValue	Adj.Pvalue
2005 *vs.* 2007	95.29502	25	3.842300e-10	5.763450e-10
2007 *vs.* 2009	33.87294	25	1.106518e-01	1.106518e-01
2005 *vs.* 2009	100.18731	25	5.834561e-11	1.750368e-10

Table 7.19: Results of Chi-squared tests of homogeneity between the distribution of frequencies of the functionalities for each pair of years.

Therefore, the Chi-squared tests of homogeneity tell us that the numbers of functionalities of GO tools are different between 2005 and 2007, as well as between 2005 and 2009, but they do not show statistical differences between 2007 and 2009. Such differences between the number of functionalities of GO tools for each year are clearly recognized in figure 7.13. In this figure there are four plots represented. The first plot (top-left) are boxplots of the number of functionalities of GO tools per year. The next three representations are scatterplots with a Loess curve and the respective confidence bands, where each one of them shows the GO tools based on the number of functionalities for a pair of years. Globally, the number of functionalities available in GO tools have been increasing over time. Looking at the interquartile ranks of boxplots, one can observe that, while in 2005 most of the GO tools show a number of functionalities varying from 22 to 42 out of 178 functionalities, in 2007 these numbers of functionalities increase, varying from 45 to 65 out of 178 functionalities, and in 2009 the number of capabilities increased again, varying from 58 to 80 out of 178 functionalities. Note that the interquartile range (IQR) seems to be wider in 2009 than in other years. The general upward trend also becomes obvious in scatterplots, whatever the combination of years being considered. This trend tends to be linear, that is, for a given year (i.e. x-axis), GO tools with a low number of functionalities, also show a low number of functionalities in the year thereafter (i.e. y-axis), and GO tools with a high number of functionalities, also show a high number of functionalities in the year thereafter. In some sense, these results suggest that promoters offering GO tools with more capacities invested much more effort in the software than the promoters that offer GO tools with less capabilities.

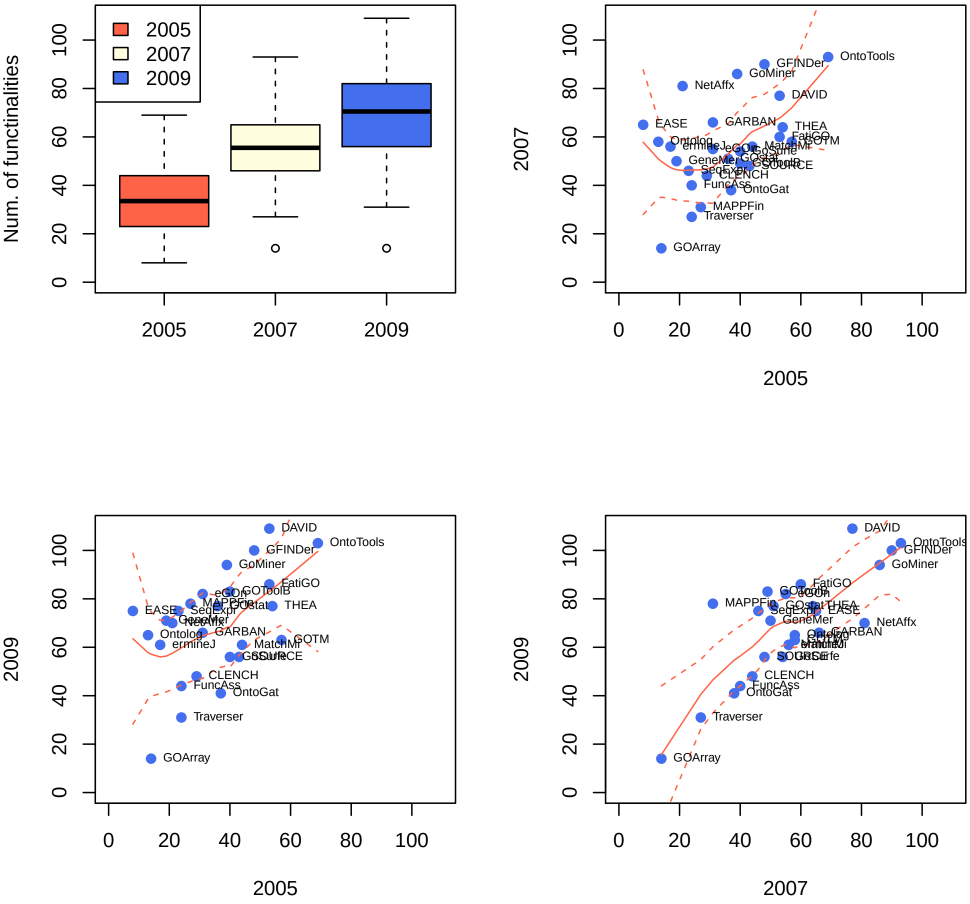


Figure 7.13: Boxplots of the number of functionalities of GO tools per year and scatterplots of the number of functionalities between each pair of years.

Analysis by Sections

Table 7.20 shows the results of Chi-squared tests of homogeneity disaggregated by sections of functionalities. Most of the tests are not statistically significant because adjusted p-values are less than a level of significance of $\alpha = 0.05$. In consequence, there is not enough evidence to reject the null hypothesis associated with each of these comparisons. However, the comparisons between years 2005 *vs.* 2007 and 2005 *vs.* 2009 in section *Supported Species*, and the comparisons between all the pairs of years in section *Annotations*, the adjusted p-values are less than a level of significance of $\alpha = 0.05$. Therefore, in these cases, we reject the null hypothesis that the distribution of frequencies between each pair of years is the same. Therefore, we reject the null hypothesis that the distribution of frequencies between each pair of years mentioned, in these sections, is the same.

Boxplots and scatterplots associated with each section of functionalities are shown in figures from 7.14 to 7.19. The overall trend of boxplots display an increase in the number of functionalities of GO tools. Note that the boxes show much higher interquartile ranks in 2009 than in previous years. However, in *Statistical Methods* and *Outputs* sections it is more difficult to appreciate clear differences in the shapes of the boxplots between years 2007 and 2009. When looking at scatterplots such a general upward trend does not become obvious. It seems to suggest that in all sections a “linear” increase exists between 2007 and 2009. But, the point clouds of scatterplots representing the number of functionalities between 2005 and 2007, and also between 2005 and 2009, seem to show fuzzy circular shapes. What is clear is that in *Supported Species* and *Annotations* sections, the point clouds of the scatterplots between 2005 and 2007, and 2005 and 2009 show a significant change in the numbers of functionalities.

Section of Functionalities	Comparison	Chi.Squared	df	PValue	Adj.Pvalue
Type Of Tool	2005 vs 2007	6.1382	25	0.9999	0.9999
	2007 vs 2009	7.3768	25	0.9998	0.9999
	2005 vs 2009	10.5671	25	0.9948	0.9999
Supported Species	2005 vs 2007	57.5300	24	0.0001	0.0004
	2007 vs 2009	14.0828	24	0.9448	0.9448
	2005 vs 2009	55.6518	24	0.0003	0.0004
Input Data	2005 vs 2007	26.6269	25	0.3748	0.5622
	2007 vs 2009	10.0341	25	0.9966	0.9966
	2005 vs 2009	34.5545	25	0.0966	0.2897
Annotations	2005 vs 2007	54.4073	25	0.0006	0.0009
	2007 vs 2009	49.2869	25	0.0026	0.0026
	2005 vs 2009	58.3251	25	0.0002	0.0005
Statistical Methods	2005 vs 2007	15.3798	23	0.8805	0.9992
	2007 vs 2009	7.3534	23	0.9992	0.9992
	2005 vs 2009	21.1231	23	0.5736	0.9992
Outputs	2005 vs 2007	16.9427	25	0.8839	0.9999
	2007 vs 2009	3.3997	25	0.9999	0.9999
	2005 vs 2009	16.9239	25	0.8845	0.9999

Table 7.20: Results of Chi-squared tests of homogeneity between the distribution of frequencies of the functionalities for each pair of years by section .

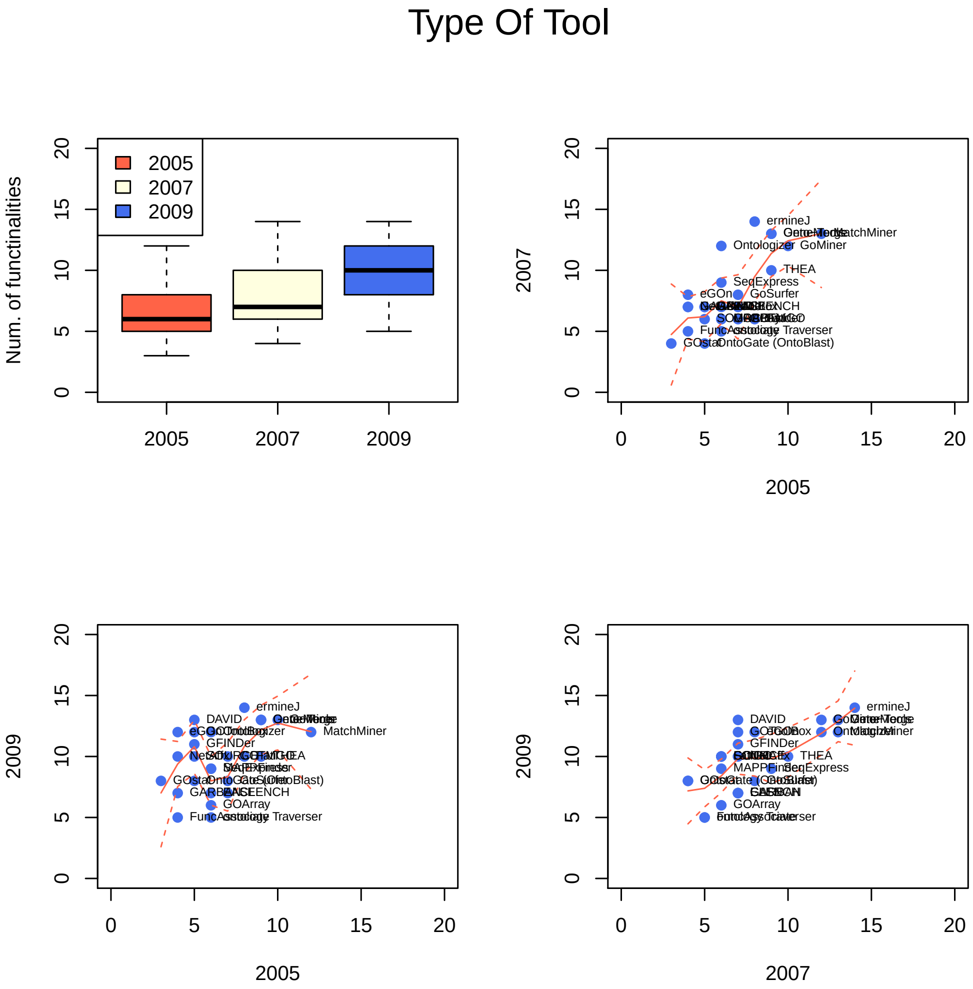


Figure 7.14: Boxplots and scatterplots of the number of functionalities of GO tools in section *Type of Tools*.

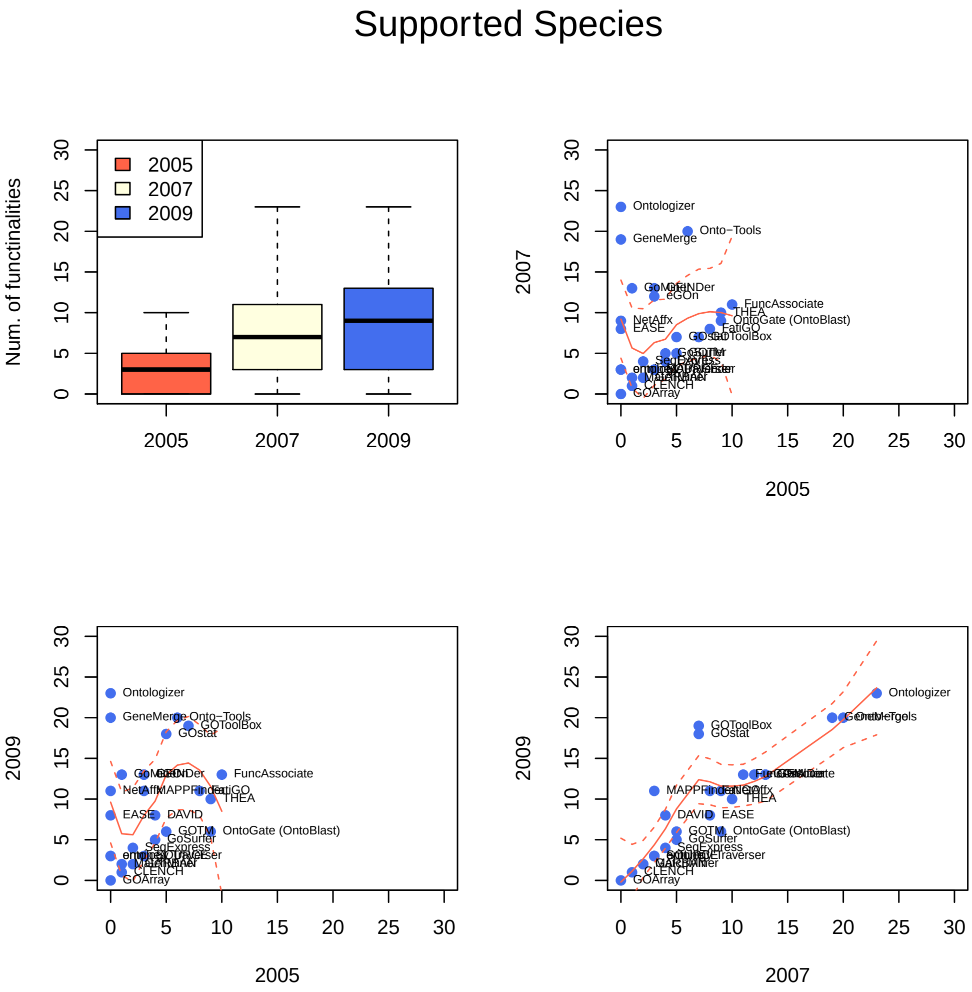


Figure 7.15: Boxplots and scatterplots of the number of functionalities of GO tools in section *Supported Specie*.

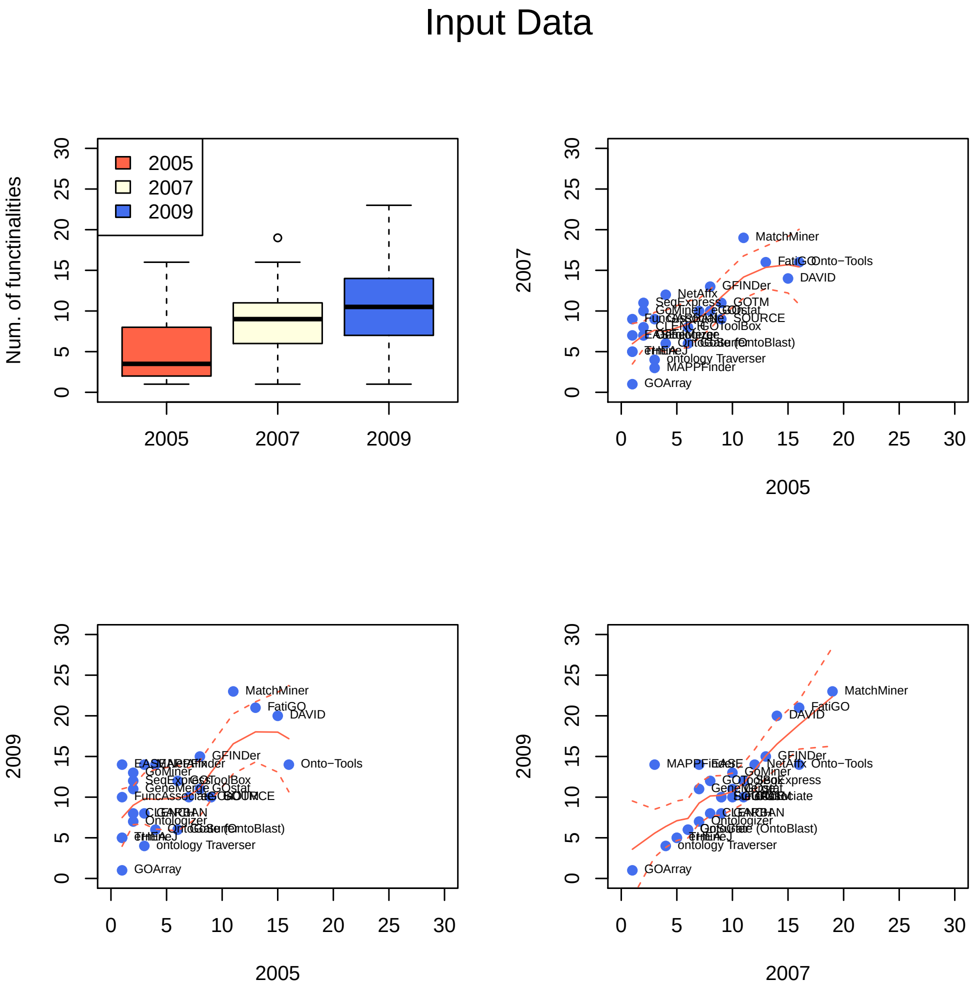


Figure 7.16: Boxplots and scatterplots of the number of functionalities of GO tools in section *Input Data.*

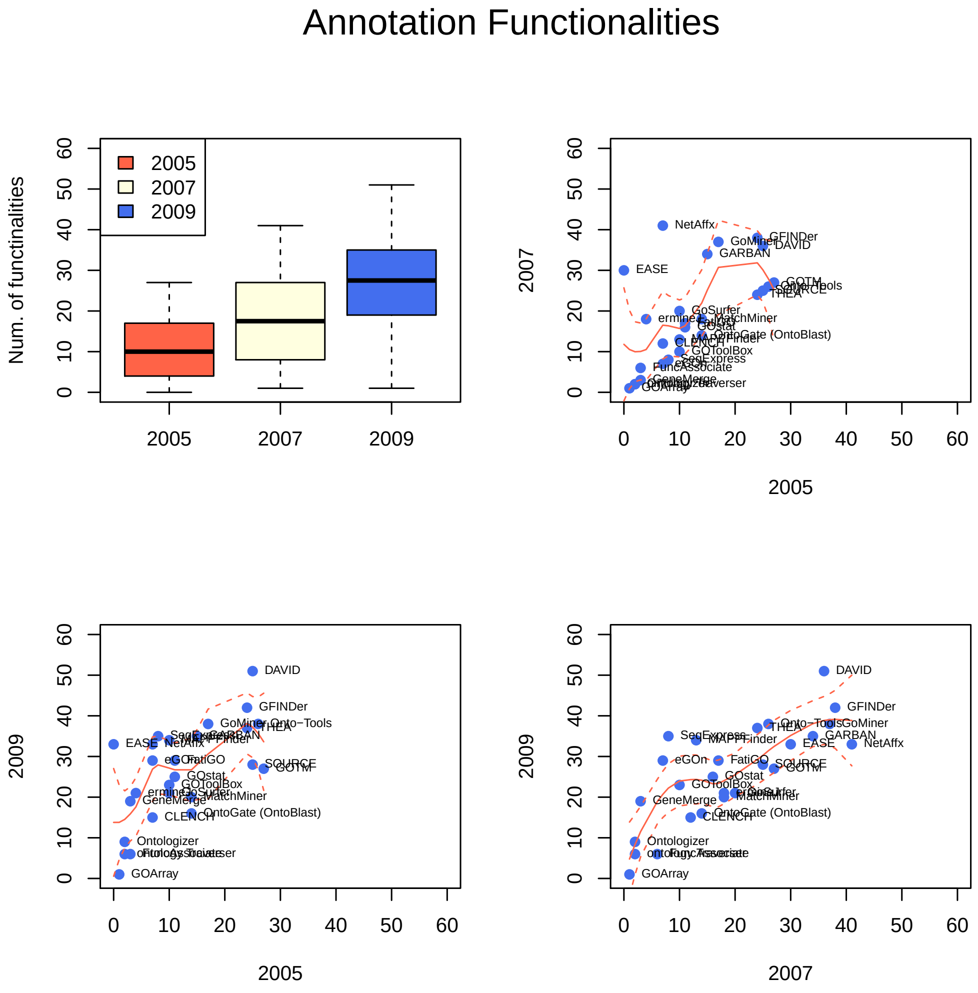


Figure 7.17: Boxplots and scatterplots of the number of functionalities of GO tools in section *Annotation Functionalities*.

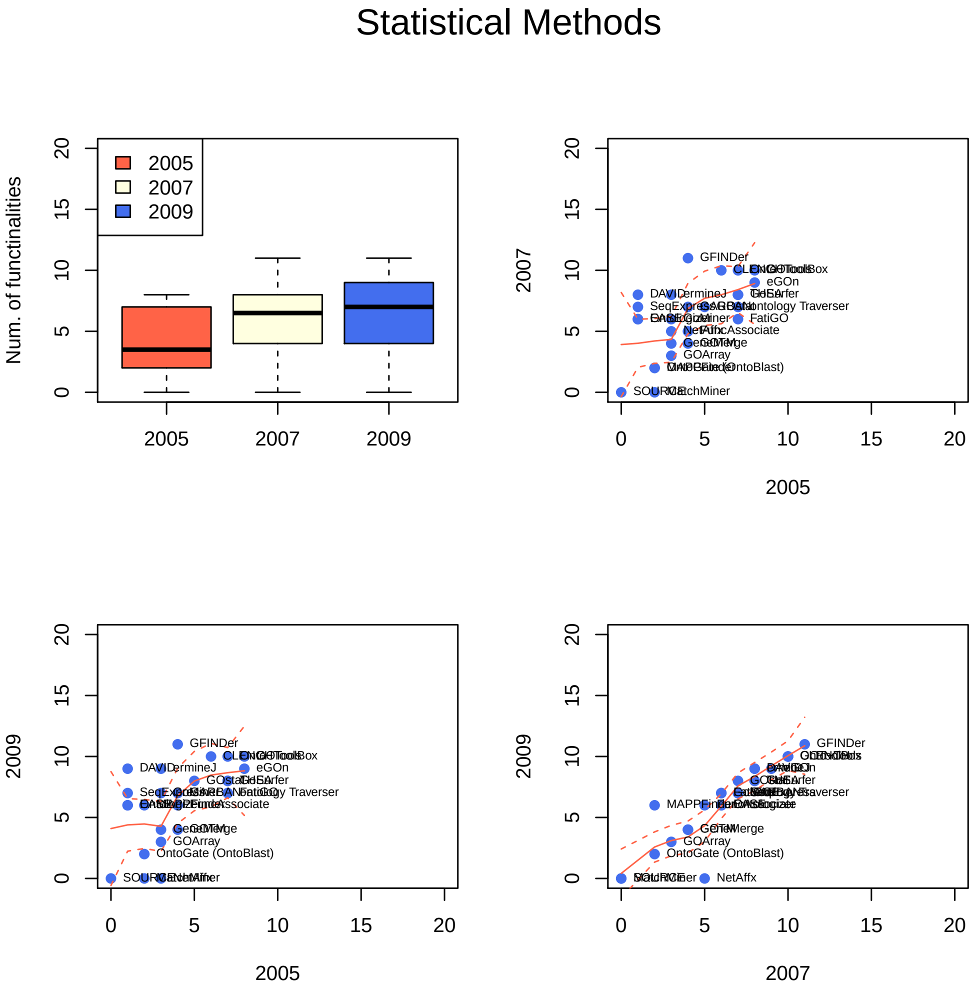


Figure 7.18: Boxplots and scatterplots of the number of functionalities of GO tools in section *Statistical Methods*.

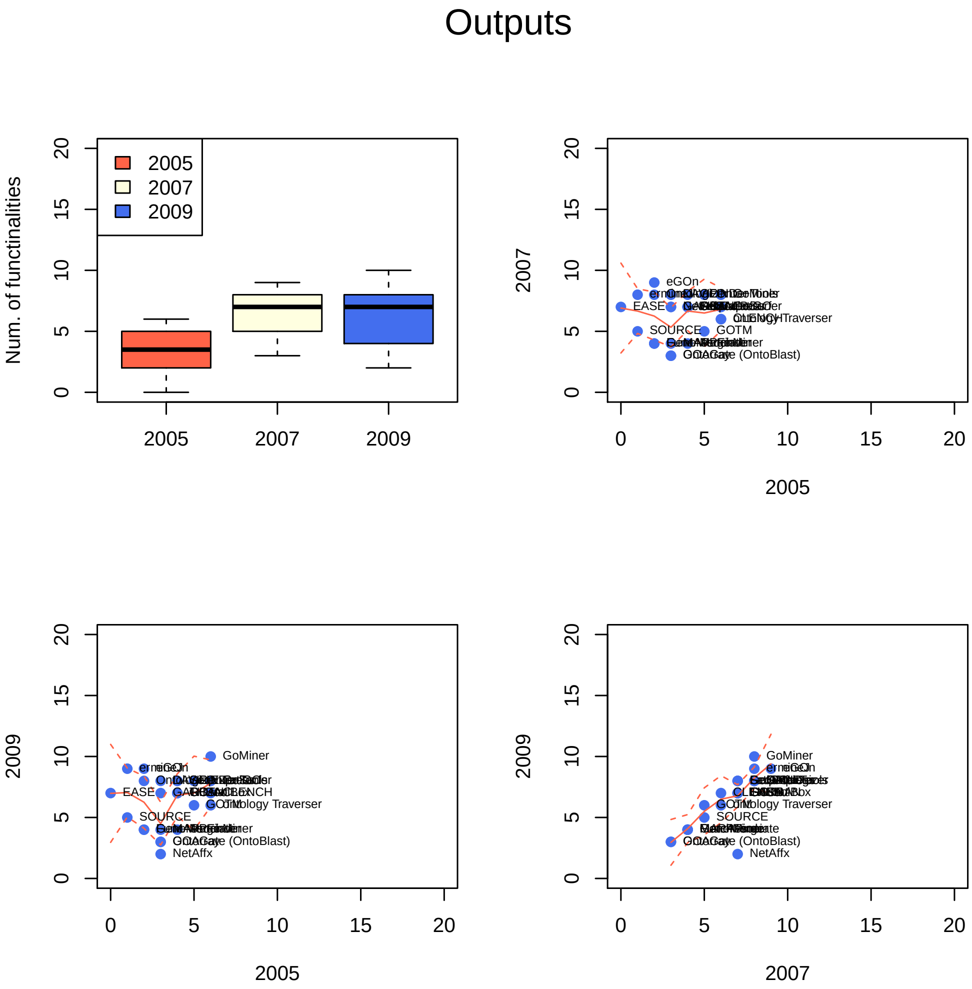


Figure 7.19: Boxplots and scatterplots of the number of functionalities of GO tools in section *Outputs*.

7.4.3 Multivariate Analysis

In this subsection, for the sake of convenience in consulting the results, some results are going to be outlined in a different order order to the order in which the associated methods have been introduced in the previous chapter (see section 6.4.2.3). Firstly, results associated with the determination of the number of clusters are presented. Secondly, hierarchical cluster results are shown. And thirdly, multidimensional scaling results are explained.

Selection of the Number of Clusters

Table 7.21 shows a summary of the optimal number of clusters according to the corresponding Silhouette Coefficients based on the Jaccard and Matching coefficients, and after running PAM algorithms. Globally speaking, in all cases there is a clear lack of substantial cluster structures. That is, due to the fact that all Silhouette Coefficients (SC) are lower than 0.25, this suggests that GO tools might not be well characterized by (if so) models determined according to the clusterings based on the dissimilarity matrices $\mathbf{D}^c_{2005}$, $\mathbf{D}^c_{2007}$ and $\mathbf{D}^c_{2009}$. In 2005, the number of clusters seems to be low (i.e. 2 and 3). In 2007, the number of clusters seems to be high (i.e. 9 and 16). And, in 2009, it is not clear because it depends on the coefficient used. Based on the Jaccard coefficient, SC suggests 2 clusters and based on the Matching coefficients it seems to be 5. It is difficult to interpreted, but it might be associated in some sense with the improvements introduced in GO tools during the period from 2005 to 2007, in order to fill a “gap” of capabilities offered to perform the enrichment analysis.

Year	Jaccard Coefficient		Matching Coefficient	
	Num. Clusters	SC	Num. Clusters	SC
2005	2	0.13	3	0.2
2007	9	0.09	16	0.1
2009	2	0.1	5	0.12

Table 7.21: Summary table of the optimal number of clusters according to the SCs based on the Jaccard and Matching Coeffients.

Bar diagrams of the average silhouette widths for each number of clusters and the silhouette plots of the optimal number of clusters per year and based on the Jaccard and Matching coefficients are shown in figures 7.20

and 7.21.

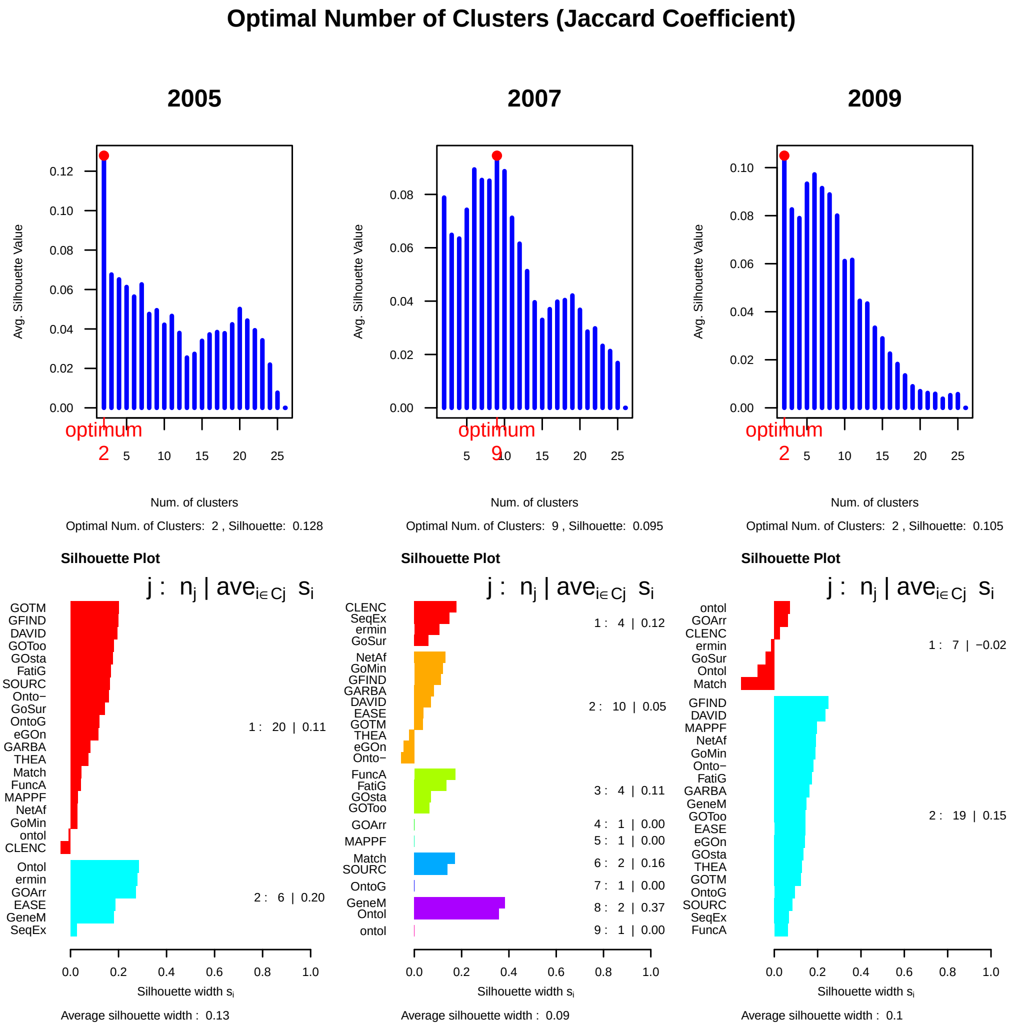


Figure 7.20: Bar plots of the average silhouette widths for each number of clusters and the silhouette plots for the optimal number of clusters per year and based on the Jaccard Coefficient.

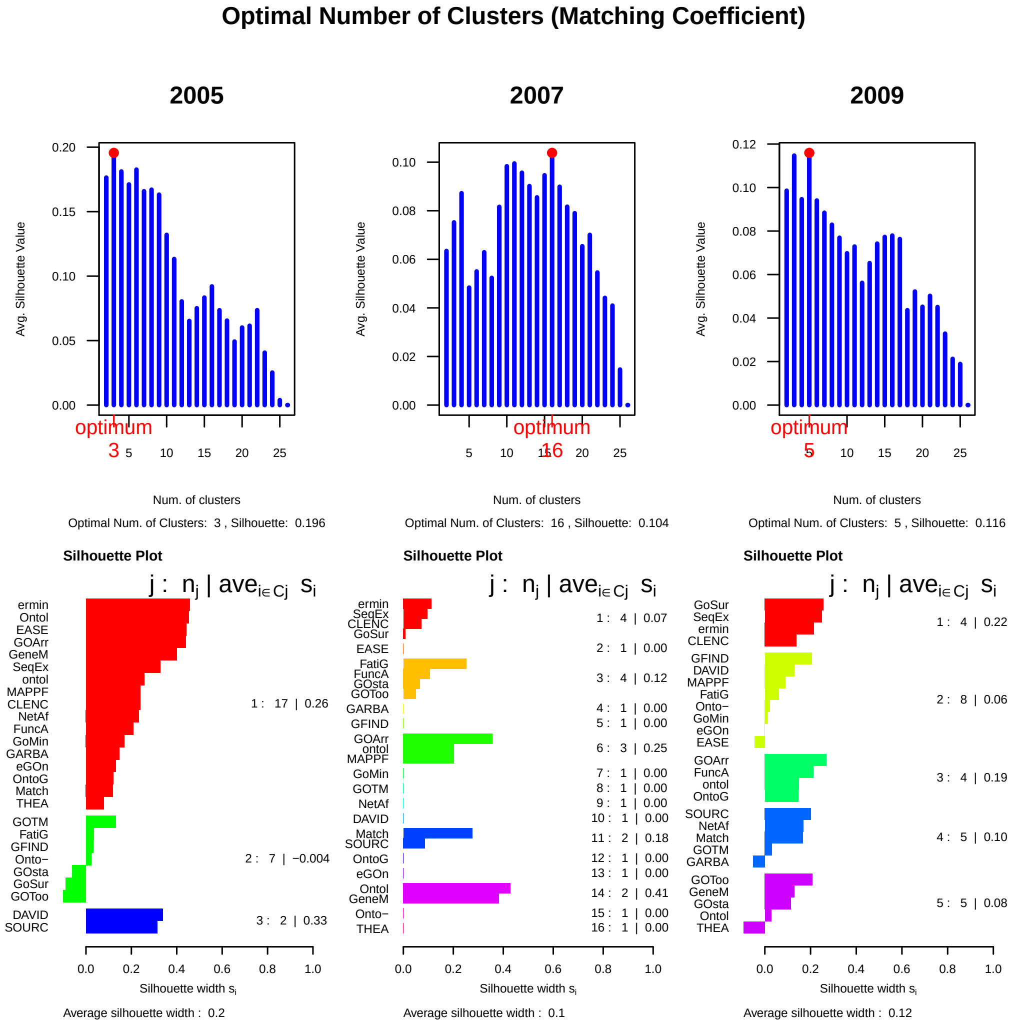


Figure 7.21: Bar plots of the average silhouette widths for each number of clusters and the silhouette plots for the optimal number of clusters per year and based on the Matching Coefficient.

Looking at the bar diagrams, whatever the year and the coefficient used, and leaving aside any limitations due to low values of SC, partitioning the space with a low number of clusters generally seems to explain a considerable part of the information about the groups of GO tools. Based on this fact and in order to discuss the evolution of GO tool functionalities, 3 clusters have been considered for highlighting groupings in hierarchical clusters and MDS.

Hierarchical Clusters

Hierarchical Clusters and Multidimensional Scaling plots of GO tools in 2005, 2007 and 2009 are shown in figures 7.22, 7.23, and 7.24, respectively. Each figure consists of six plots divided into two rows. The first row shows the hierarchical cluster, the classical MDS and the non-metric MDS based on the Jaccard Coefficient. The second row shows the same plots, but based on the Matching Coefficient. GO tool names in dendrograms and points corresponding to GO tools in both MDS plots are colored in gold, red or blue. Each of these colors is associated with one of the three clusters that have been considered after running the PAM algorithm several times and checking the Silhouette Coefficients associated with each similarity coefficient and year.

Globally, hierarchical clusters show a large number of GO tools lying in one "major" cluster (red) and the remain GO tools are divided between two "minor" clusters (gold and blue), regardless of both the year and coefficient of similarity used. More specifically, hierarchical clusters based on the Jaccard coefficient clearly show most of the GO tools lying in the major cluster (red) and the minor clusters (gold and blue) have very few GO tools. In 2005, the major cluster consists of 20 GO tools, and the minor clusters contain 5 (gold) and 1 (blue) GO tools. These quantities are quite similar to the number of GO tools lying in the three clusters in 2007 and 2009, because the number of GO tools lying in the major cluster increases to 24 programs in both 2007 and 2009, and consequently the number of GO tools lying in the minor clusters are reduced, leaving one GO tool in one cluster (gold) and another GO tool in the other cluster (blue) in both years. However, this behavior is quite different when looking at the hierarchical clusters based on the Matching coefficient. Although most of the GO tools lie in a major cluster (red), this fact is not as obvious as with the hierarchical clusters based on the Jaccard coefficient. While in 2005 the major cluster consists of almost all GO tools, in 2007 the cluster experienced a drastic decrease in its number of GO tools, which obviously causes an increase in the number of GO tools in the minor clusters, and this trend is maintained in 2009. Specifically, in 2005, the major cluster consists of 23 GO tools, and the minor clusters have 2 GO tools (gold) in one case and 1 GO tool (blue) in the other case. In 2007, the major cluster consists of 16 GO tools, and the minor clusters contain 8 GO tools (gold) in one case and 2 GO tools (blue) in the other case. Finally, in 2009, the major cluster is formed by 14 GO tools (red), the second cluster consists of 8 GO tools (gold) and in the third cluster there are 4 programs (blue). Table 7.22 shows the numbers of clusters assigned to each GO tool per year and the coefficient of similarity, after “cutting” the corresponding hierarchical cluster.

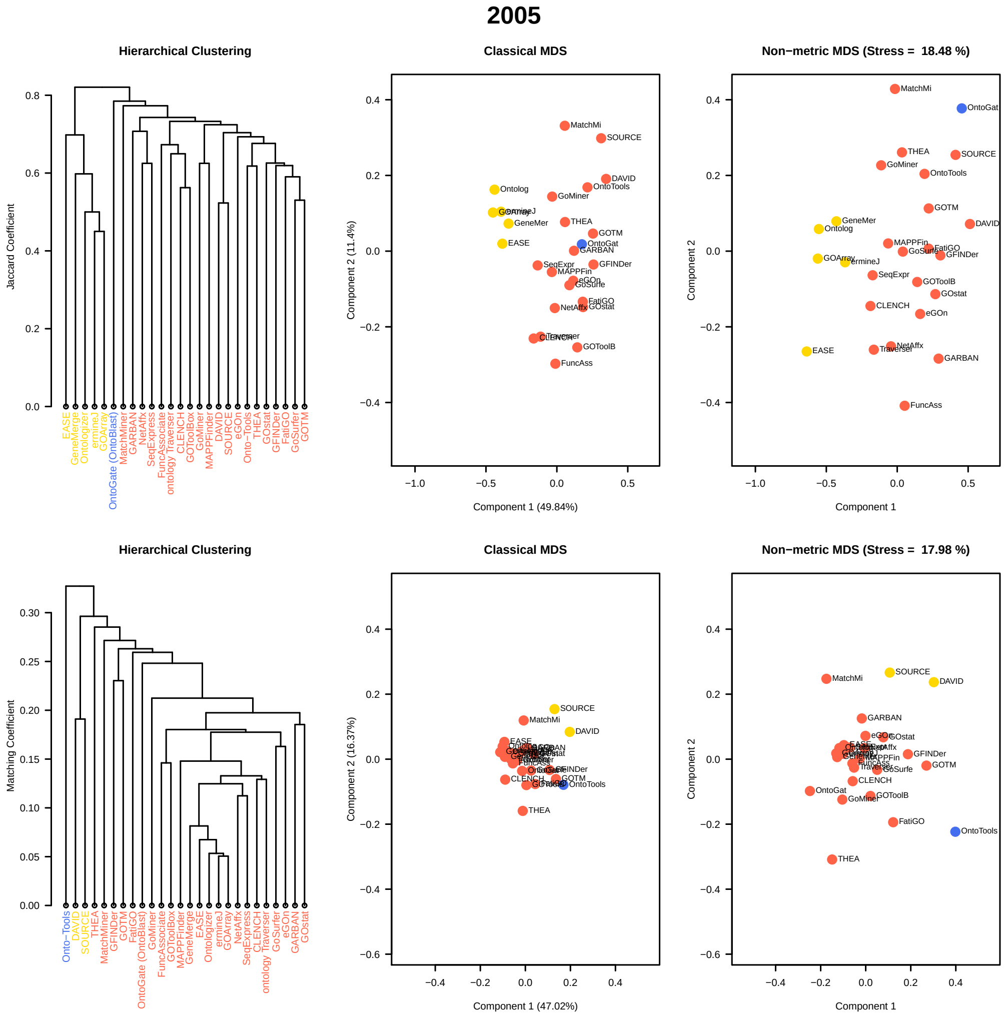


Figure 7.22: Hierarchical clusters and two-dimensional plots of MDS solutions based on the Jaccard and Matching Coefficients of GO tools in 2005

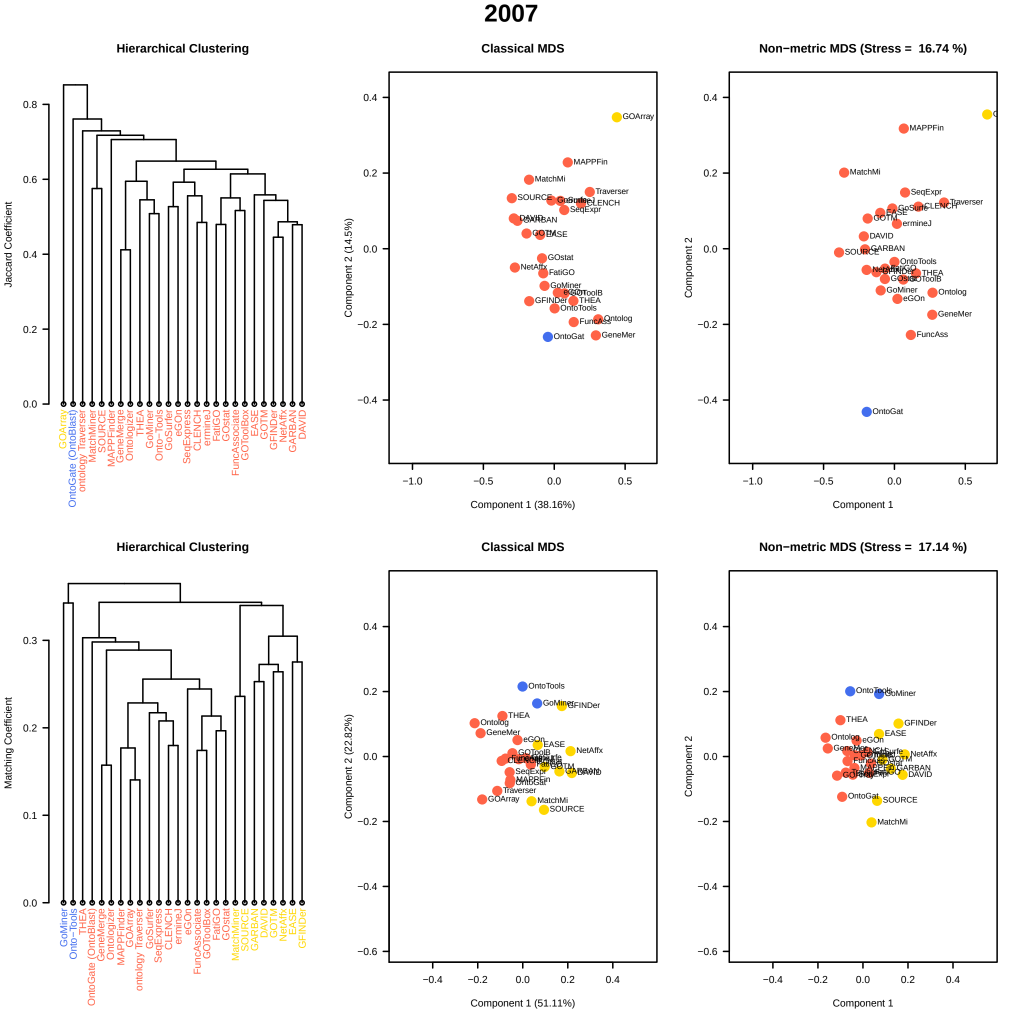


Figure 7.23: Hierarchical clusters and two-dimensional plots of MDS solutions based on the Jaccard and Matching Coefficients of GO tools in 2007

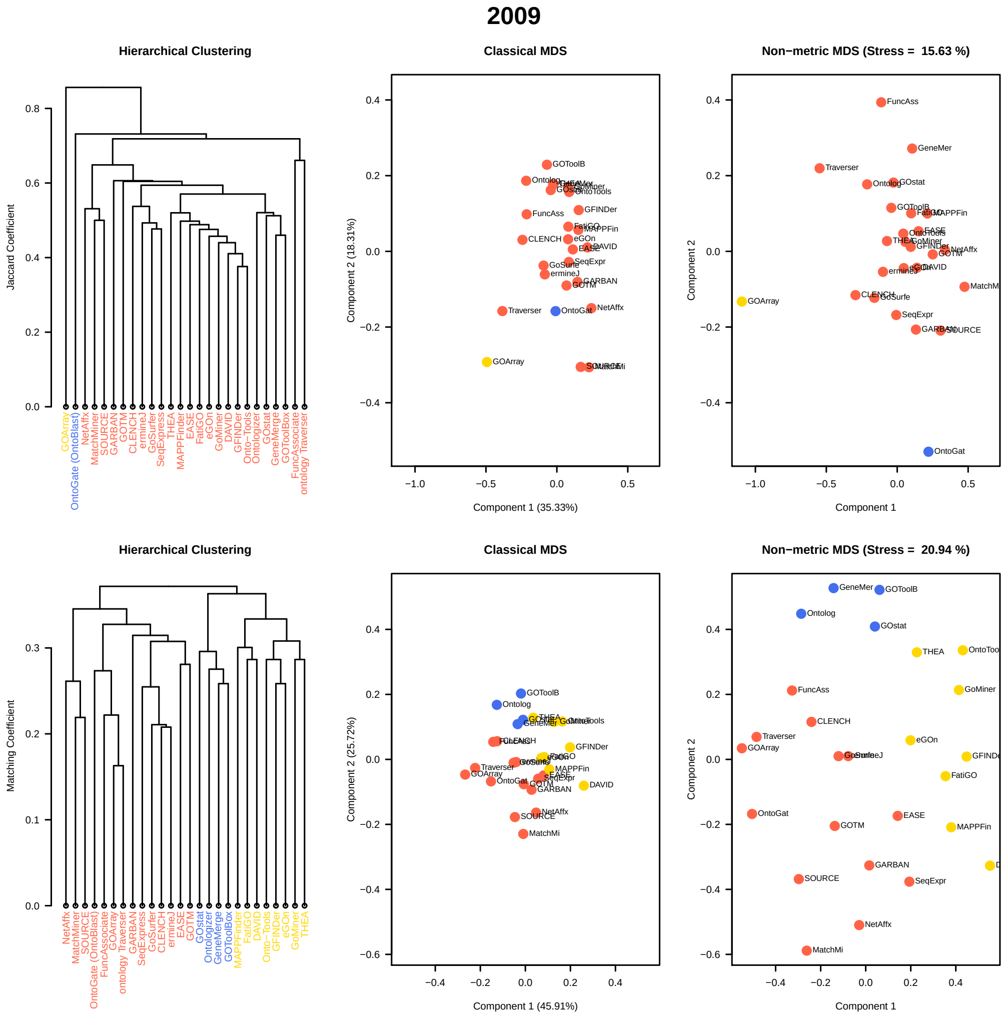


Figure 7.24: Hierarchical clusters and two-dimensional plots of MDS solutions based on the Jaccard and Matching Coefficients of GO tools in 2009

Such differences between hierarchical clusters based on the Jaccard coefficient and Matching coefficients are obviously attributable to the way of understanding and defining each coefficient. That is, the Jaccard coefficient only considers positives matches (i.e. it counts functionalities that are available in both GO tools), and the Matching coefficient considers positive and negative matches (i.e. it counts both functionalities that are available and not available in both GO tools) 6.2. When considering the Jaccard coefficients the idea of the presence of certain homogeneity among GO tools over time does of course arise. Note that this fact is in consonance with the lack of cluster structures suggested by the silhouette coefficients (i.e. 0.13 in 2005, 0.09 in 2007, and 0.1 in 2009). However, when considering the Matching coefficient, GO tools seems to evolve from the homogeneity of capabilities to a some sort of "specialization" of programs. As stated above, lacks of cluster structures is shown by the silhouette coefficients (i.e. 0.2 in 2005, 0.1 in 2007, and 0.12 in 2009) too, but in all cases they are slightly higher than those average silhouettes widths based on the Jaccard coefficient. In any case, the three clusters selected based on the Matching coefficient show a subset of GO tools that always "go" together (i.e. `CLENCH` ([137]), `ermineJ`, `FuncAssociate` ([16]), `GOArray`, `GoSurfer` ([169]), `OntoGate (OntoBlast)` ([167]), `ontology Traverser` ([164]), and `SeqExpress` ([19])). Note that these GO tools are those points that lie approximately on the imaginary line bisecting the scatterplots in figure 7.13. That is, those points that are approximately "invariant" over time, or in another sense those points representing GO tools that do not seem to have experienced major changes with regard to their functionalities.

GO Tool	Jaccard Coefficient			Matching Coefficient		
	2005	2007	2009	2005	2007	2009
CLENCH	1	1	1	1	1	1
EASE	2	1	1	1	2	1
ermineJ	2	1	1	1	1	1
FatiGO	1	1	1	1	1	2
FuncAssociate	1	1	1	1	1	1
GARBAN	1	1	1	1	2	1
GFINDer	1	1	1	1	2	2
GOArray	2	2	2	1	1	1
GoMiner	1	1	1	1	3	2
GOstat	1	1	1	1	1	3
GoSurfer	1	1	1	1	1	1
GOTM	1	1	1	1	2	1
MAPPFinder	1	1	1	1	1	2
NetAffx	1	1	1	1	2	1
DAVID	1	1	1	2	2	2
MatchMiner	1	1	1	1	2	1
OntoGate (OntoBlast)	3	3	3	1	1	1
SOURCE	1	1	1	2	2	1
eGOn	1	1	1	1	1	2
GeneMerge	2	1	1	1	1	3
GOToolBox	1	1	1	1	1	3
Onto-Tools	1	1	1	3	3	2
Ontologizer	2	1	1	1	1	3
ontology Traverser	1	1	1	1	1	1
SeqExpress	1	1	1	1	1	1
THEA	1	1	1	1	1	2

Table 7.22: Number of the cluster assigned to each GO tool per year and coefficient of similarity.

Multidimensional Scaling

The 2-dimensional plots of MDS solutions are shown in figures 7.22, 7.23, and 7.24. Each point in one of these plots is associated with one GO tool and is coloured red, gold or blue according to the cluster in which it lies.

A general overview of the plots by years suggests that GO tools are more spread out in MDS solutions based on the Jaccard coefficient than in MDS solutions based on the Matching coefficient, independently of the type of approach used (i.e. classical or non-metric MDS). There are some subtle differences in "meaning", but globally speaking, from the non-metric MDS solution based on the Matching coefficient in 2009, distances among points seem to behave similarly.

Based on the Jaccard coefficient, for all years, classical MDS solutions show values ranging from -0.5 to 0.5 on the first dimension and from -0.4 to 0.4 on the second dimension. The adequacy for each year are 61.23% in 2005, 52.67% in 2007, and 53.64% in 2009. These percentages suggest that representation of data by the first two dimensions is not bad. Clusters of GO tools that have been determined with the help of silhouettes 7.4.3 and identified in dendrograms 7.4.3, do not show an outstanding separation effect among them. However, in 2005 the MDS plot seems to suggests that one of the minor clusters (gold) is slightly separated to the left from any other GO tool, and one of the programs of such a cluster (i.e. `GOArray`) remains a little bit separated from any other GO tool in 2007 and 2009. In the case of non-metric MDS solutions the behavior is more or less the same as in classical MDS solutions. On the first dimension, points associated with GO tools range from -0.5 to 0.5 in 2005 and 2007, but in 2009 values vary from -1 to 0.5. On the second dimension, points range from -0.4 to 0.4 in 2005 and 2007, but in 2009 this interval is higher, varying from -0.6 to 0.4. Stress values associated with each year are 18.48% in 2005, 16.74% in 2007 and 15.63% in 2009, that *a priori* suggests poor or fair goodness of fit, which will be discussed later on.

Based on the Matching coefficient, for all years, points of classical MDS solutions range approximately from -0.2 to 0.2 on the first dimension, as well as on the second dimension. But, note that as years go by this interval is being spread little by little until it ranges from -0.3 to 0.3, more or less. The measures of agreement for each year are 63.39% in 2005, 73.93% in 2007, and 71.63% in 2009. These percentages suggest that representation of data by the first two dimensions is much better than classical MDS solutions based on Jaccard coefficient. Clusters of GO tools that have been determined do not show again an outstanding separation effect between them, but in contrast to MDS solutions based on the Jaccard coefficients, for all years both approaches (i.e. classical and non-metric MDS) seem to place the clusters of GO tools in the same areas. Of course, this vague assertion is not free of criticism, but it could be confirmed considering a higher dimensionality. In any case, what is clear is that in classical MDS solutions based on the Matching coefficient, clusters are less fuzzy and points are more crowded together than in classical MDS solutions based on the Jaccard coefficient. In the case of non-metric MDS solutions the behavior in 2005 and 2007 is more or less the same as in classical MDS solutions, but completely different in 2009. On the first dimension, in 2005, points vary from -0.3 to 0.4, in 2007 the values range from 0.2 to 0.2, and in 2009 the points are lying in a higher interval ranging from -0.5 to 0.5. On the second component of dimensionality, in 2005 the values range from -0.3 to 0.4, in 2007 the points vary from -0.2 to 0.2, and in 2009 the values range from -0.6 to 0.5. That is, in 2007 there is apparently a contraction of points representing the GO tools, to be followed by an expansion in 2009. This fact has no easy interpretation, and even more so if the stress values are taken into account, because they are 17.98% in 2005, 17.14% in 2007, and 20.94% in 2009. A possible explanation may be that it relies on the specialization process over the time that we have observed in descriptive statistics 7.4.1 and inferential analysis 7.4.2, where results suggest that in 2007 most of the promoters introduced and improved the capacities of GO tools considerably. However, while progress continued until 2009, the results suggest that in parallel a brake began to be automatically and significantly triggered. It therefore seems that the two-dimensional plots of the non-metric MDS show a kind of graphical description regarding the evolution and specialization of GO tools, but which in turn is not clear enough to reflect the changes in terms of distances and clusters due to the fact that the stress values suggest a poor goodness of fit.

Adequacy and Stress of Multidimensional Scaling Solutions

Figure 7.25 shows bar diagrams of adequacy associated with each classical MDS solution based one the Jaccard coefficient (first row) and the Matching coefficient (second row). For each bar diagram two points are highlighted in red and orange. The red point indicates the accumulated percentage of adequacy associated with dimension 2 and the orange point indicates the accumulated percentage of adequacy associated with dimension 3.

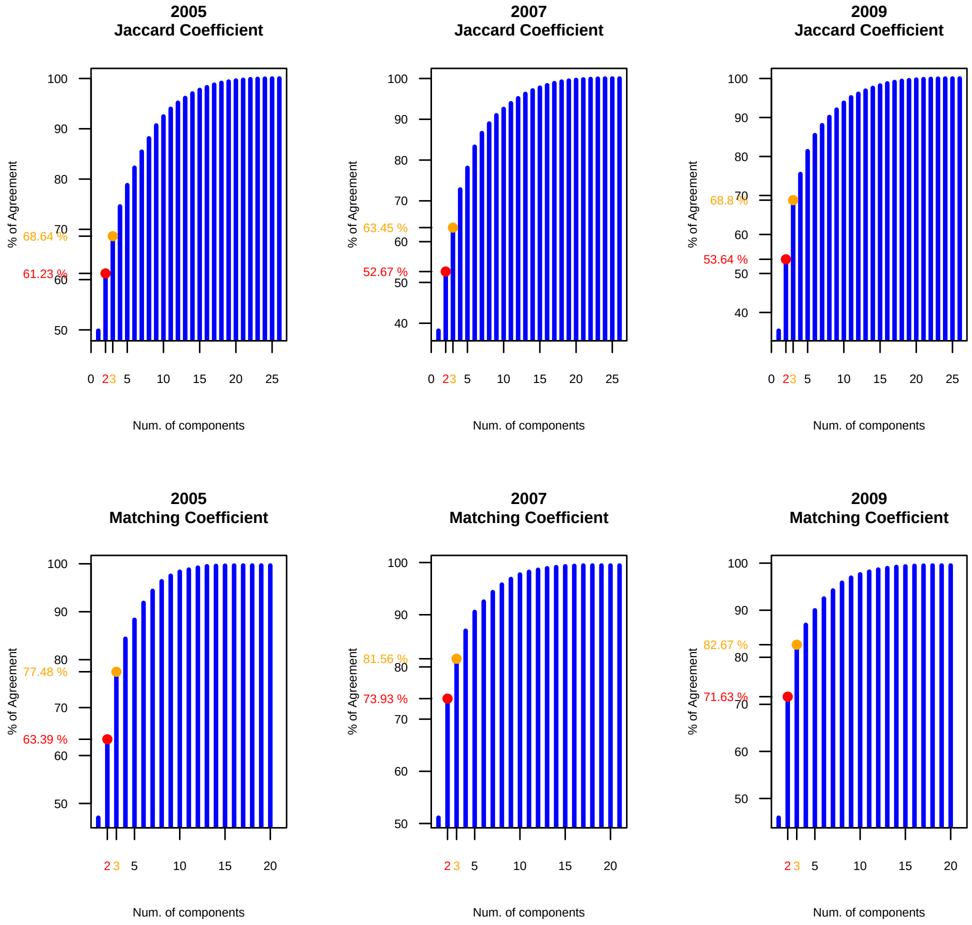


Figure 7.25: Adequacy plots associated with MDS solutions.

Interpretation of this kind of bar diagram is quite similar to the interpretation of the scree plot ([18], [69]) for non-metric MDS solutions 6.4.2.3, because in essence it is the same. However, in this case, the higher the dimension selected, the greater the percentage of agreement accumulated. The optimal dimension that should be selected is the site where there is a "sudden change" of the curve, in the sense that new components do not contribute with significant improvements. In most of the cases of these plots they do not seem to show abrupt changes in the shapes of the curves. Thus, as mentioned in the previous paragraph 7.4.3, by considering the two-dimensional configuration, no bad representations are reached.

Scree plots and Shepard diagrams associated with non-metric MDS solutions based on the Jaccard coefficient are shown in figure 7.26, and are shown in figure 7.27 based on the Matching coefficient. In both figures, the first row shows scree plots and the second row shows Shepard diagrams. *A priori* these plots show some discrepancies that might be misinterpreted.

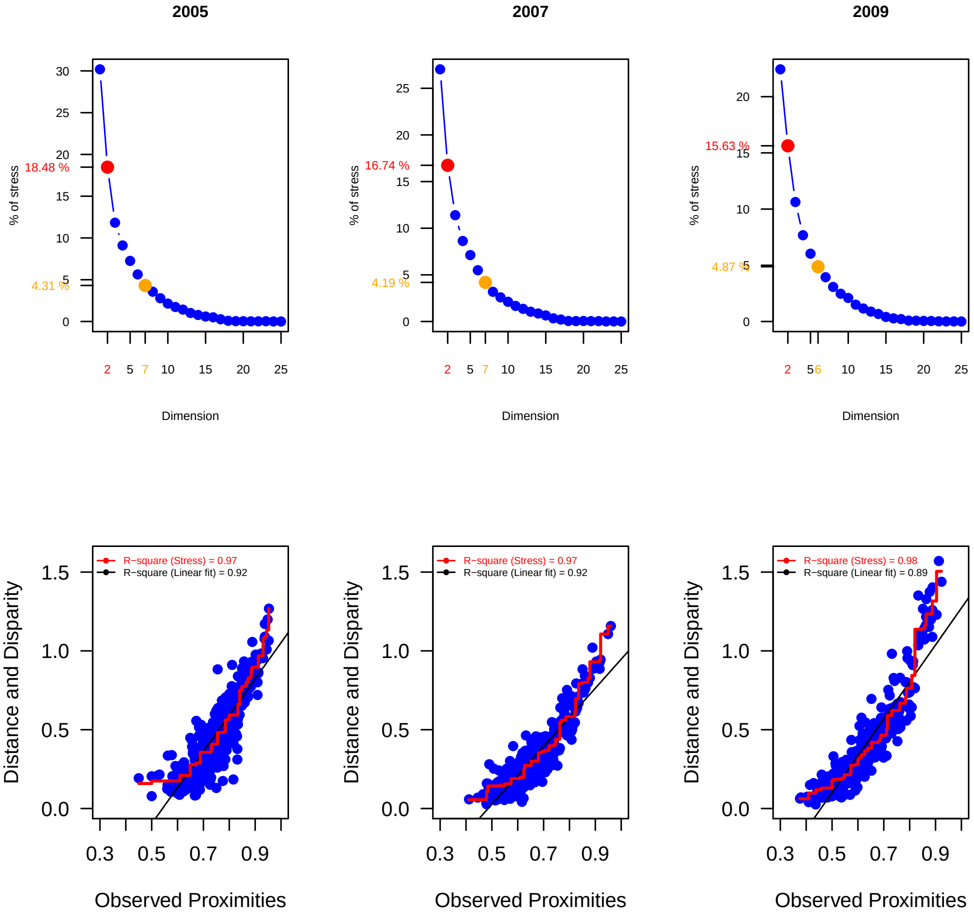


Figure 7.26: Scree plots and Shepard diagrams associated with MDS solutions based on the Jaccard coefficient.

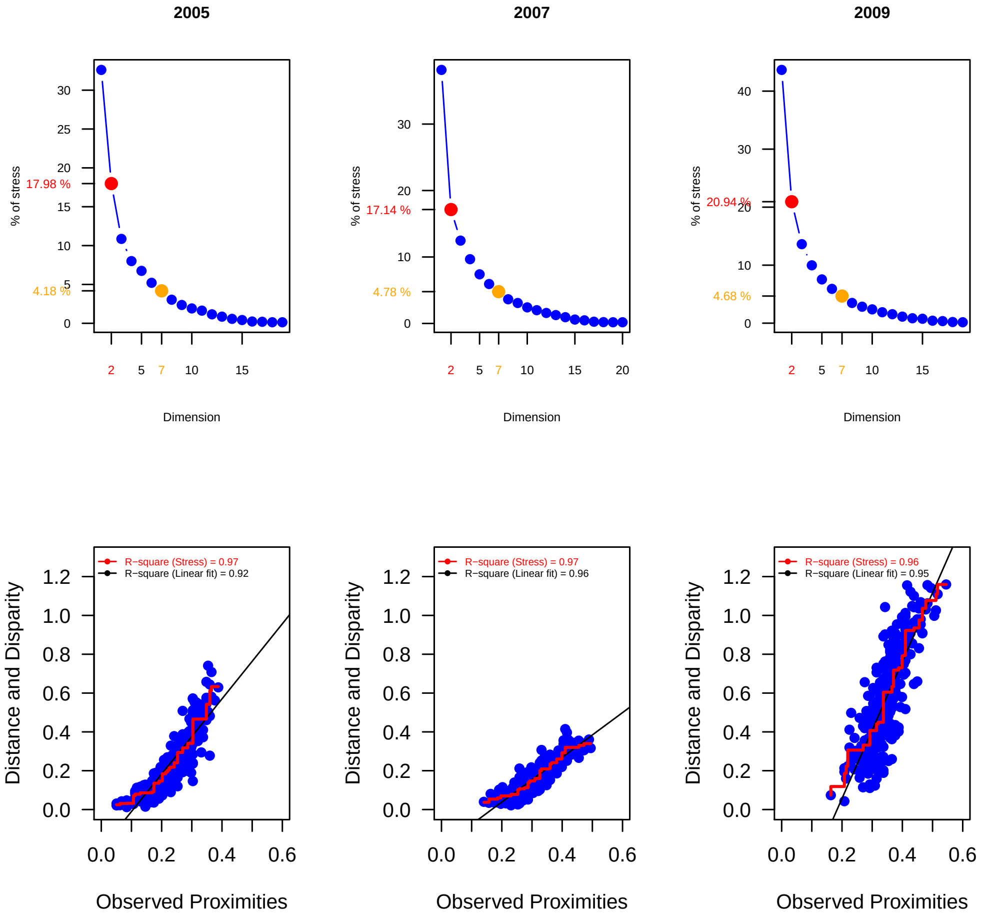


Figure 7.27: Scree plots and Shepard diagrams associated with MDS solutions based on the Matching coefficient.

In Scree plots the percentages of stress $\sigma_1(\mathbf{X})$ are represented with respect to the number of dimensions considered. In these plots there are two highlighted points, one in red and the other in orange. The first indicates the stress for dimension 2 and the second indicates the stress for the first dimension that shows a stress value lower that 5%, according to the benchmark of a "good" goodness of fit purposed by Kruskal (table 6.4). Thus, as mentioned in previous section 7.4.3, independently of the coefficient of similarity used, percentages of stress $\sigma_1(\mathbf{X})$ associated with the two-dimensional configuration of non-metric MDS solutions show fair or poor goodnesses of fit, because when looking at red points, the stress values range between 15% and 21% (red points). Note that none of the curves display an abrupt "elbow". In fact, all the curves decrease in a smoother way, although of course the decrease in stress values is much lower for higher dimensions. Therefore, by considering 5% as a cutoff for deciding the "optimal number" of dimensions, the orange points indicate that good non-metric MDS solutions should be those that consider 7-dimensional configurations for all cases with only one exception (i.e. non-metric MDS solution based on the Jaccard coefficient in 2009), which should be 5-dimensional configurations.

In Shepard diagrams, globally speaking, three main traits are observed. First, these plots display roughly linear regression curves. They show a number of marked steps, especially when looking at the diagrams associated with non-metric MDS solutions based on the Matching coefficient, and slightly curved shapes in the cases based on the Jaccard coefficient. Second, in the diagrams associated with non-metric MDS solutions based ont the Matching coefficient, we can graphically observe some differences between slopes. Third, independently of the year and the coefficient used, their coefficients of determination based on the stress (i.e. $R^2 = 1 - \sigma_1(\mathbf{X})^2$) show values higher than 0.96 (red legends at the top of each plot). In other words, the respective isotonic regressions models fit very well.

Stress values have been criticized for being over-simplistic ([160], [18]). For example, for the same underlying data structure, a larger data set will necessarily result in a higher stress value, or for instance stress values might be highly influenced by "outliers". What is clear is that, in this analysis and in keeping with previous results, scree plots suggest that the two-dimensional non-metric MDS spaces are not reliable enough, and it is therefore difficult to ensure that clusters of GO tools are identified. However, in contrast, Shepard diagram results seem to suggest that distances and disparities are good when they approximate the original proximities, that is, the distances between GO tools. Hence, a graphical identification of clusters of GO tools is apparently observed, even when the separation between such clusters is a little bit fuzzy. But note that the cloud of points in all cases are displaced from the ideal bisecting line, and this fact is not extraordinary. The points in a Shepard diagram are not, strictly speaking, geometric projections of the proximities. In fact, it is a technique to project a dissimilarity (or distance) matrix to fewer dimensions.

To sum up, coefficients of determination suggest that isotonic regression models are well represented for each 2-dimensional non-metric MDS solutions, but stress values yielded for these non-metric configurations are high, and in order to reduce such values, Scree plots suggest increasing the dimensionality on the non-metric MDS configuration, a dimension close to 7 being "optimal". So, potential models or specialization of groups of GO tools observed *a priori* are not being believable.

Mantel Tests

Table 7.23 summarizes the results yielded in Mantel and Partial Mantel Tests.

(Partial) Mantel Test	r_M	PValue
$r_M(D^J_{2005}, D^J_{2007})$	0.450	0.0001
$r_M(D^J_{2007}, D^J_{2009})$	0.803	0.0001
$r_M(D^J_{2005}, D^J_{2009})$	0.276	0.0033
$r_M(D^J_{2005}, D^J_{2009} \vert D^J_{2007})$	-0.160	0.9727
$r_M(D^M_{2005}, D^M_{2007})$	0.479	0.0002
$r_M(D^M_{2007}, D^M_{2009})$	0.588	0.0001
$r_M(D^M_{2005}, D^M_{2009})$	0.283	0.0001
$r_M(D^M_{2005}, D^M_{2009} \vert D^M_{2007})$	0.003	0.4863

Table 7.23: Mantel and Partial Mantel Tests.

The Simple Mantel Tests have been used to test the correlation between each pair of dissimilarity (distance) matrices D^c_{year} where $year = \{2005, 2007, 2009\}$ and $c = \{J, M\}$ (see section 6.4.2.3). The p-values has been determined by specifying 9999 permutations of the rows and columns of the first matrix in each case. The results indicate that all the tests are statistically significant at an α of 0.05. However, the Pearson product-moment correlation coefficients r_M indicate that there is poor correlation between each pair of dissimilarity matrices, with the exception of $r_M(D^J_{2007}, D^J_{2009})$, whose value 0.803 indicates a relatively good correlation.

The Partial Mantel Tests have been used to estimate the correlation between the two matrices D^c_{2005} and D^c_{2009}, while controlling for the effect of the matrix D^c_{2007}, and where $c = \{J, M\}$. The p-values has been determined by specifying 9999 permutations of the rows and columns of the first matrix in each case, so the correlation structure between the first and second dissimilarity matrices have been kept constant. The results indicate that, independently of the similarity coefficient used, the p-values are not statistically significant at an alpha of 0.05. Therefore, we do not have enough evidence for rejecting the null hypothesis.

In other words, whatever the coefficient of similarity used (i.e. Jaccard coefficient or Matching coefficient), the simple Mantel tests show some sorts of fair relationships between each pair of dissimilarity matrices of GO tools. This fact may suggests that the similarities between GO tools evolve over time. However, the Partial Mantel Tests, independently of the coefficient of similarity used, suggest that the matrix of dissimilarities between each pair of GO tools in 2005 do not show a linear relationship with the matrix of dissimilarities in 2009 when taking into account the matrix of dissimilarities in 2007. That is, the similarities between GO tools do not seem to be the "same" during this period of time. After considering the results, when we focus on the comparison between the dissimilarity matrices D^c_{2005} and D^c_{2009}, we might think that there is a contradiction. However, two questions must be taken into account. First, in Simple Mantel Tests the coefficients of correlations are close to 0 (i.e. $r_M(D^J_{2005}, D^J_{2009}) = 0.276$ and $r_M(D^M_{2005}, D^M_{2009}) = 0.283$), that is, the correlation is practically null, and second, in Partial Mantel Tests we are controlling for the action of a third matrix, the matrix of dissimilarities between GO tools in 2007. This third matrix is involved in removing the spurious effect of correlation that might not be seen in Simple Mantel Tests.

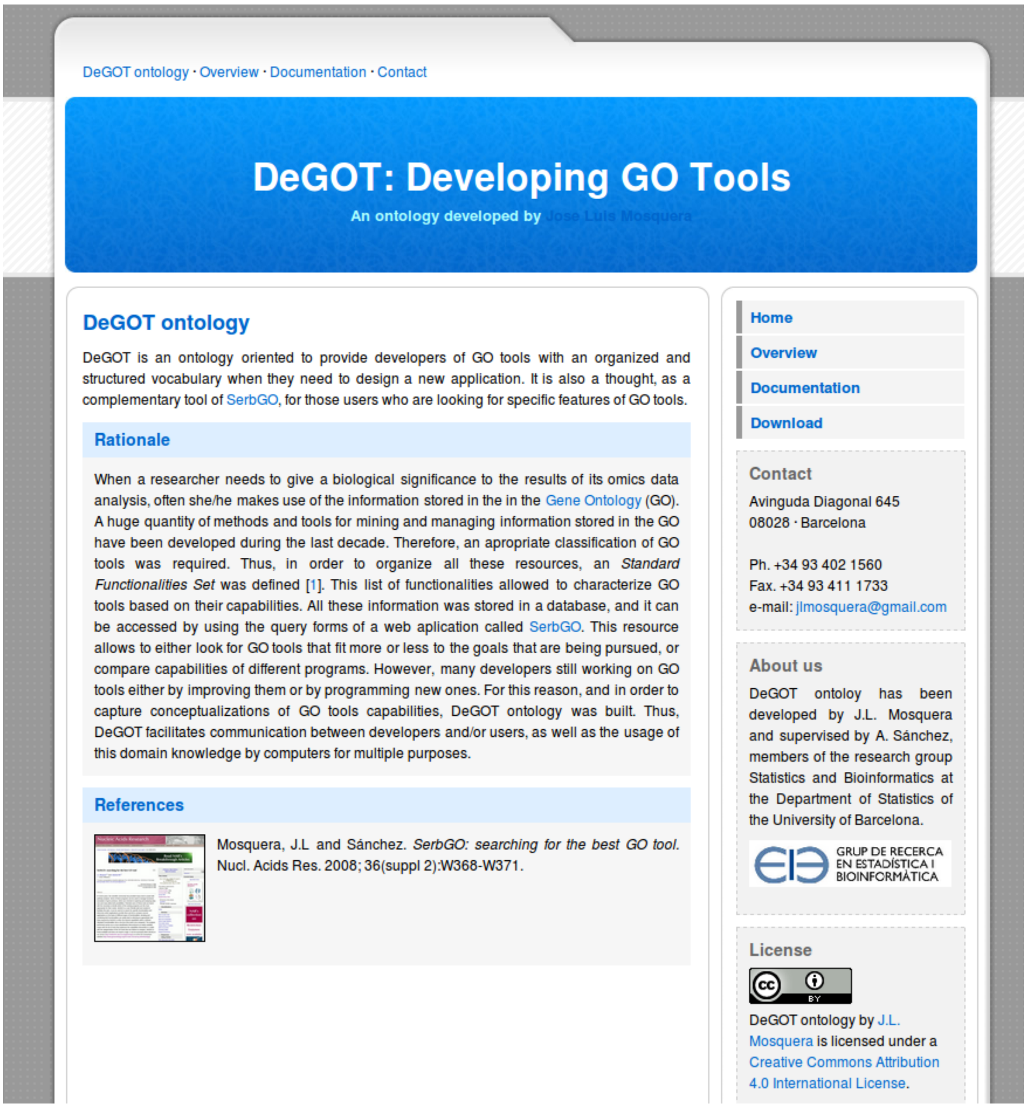
7.5 An Ontology for Developing GO Tools (DeGOT)

DeGOT is a simple ontology aimed at providing developers with an organized and structured vocabulary when they have to design a new GO tool. Furthermore, it is also a resource to help users, as a complementary tool of SerbGO, when they need to look for specific features of the GO tools.

7.5.1 Availability of DeGOT

DeGOT is freely available under a Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/) and does not require a login. It can be downloaded from the website http://estbioinfo.stat.ub.es/apli/degot of the *Statistics and Bioinformatics Research Group* (http://eib.stat.ub.edu/) lead by Dr.Alex

Sánchez. Documentation about concepts, properties and individuals included in the ontology is also available on the website. Figure 7.28 shows the main page of the DeGOT website.



DeGOT ontology · Overview · Documentation · Contact

DeGOT: Developing GO Tools

An ontology developed by Jose Luis Mosquera

DeGOT ontology

DeGOT is an ontology oriented to provide developers of GO tools with an organized and structured vocabulary when they need to design a new application. It is also a thought, as a complementary tool of SerbGO, for those users who are looking for specific features of GO tools.

Rationale

When a researcher needs to give a biological significance to the results of its omics data analysis, often she/he makes use of the information stored in the in the Gene Ontology (GO). A huge quantity of methods and tools for mining and managing information stored in the GO have been developed during the last decade. Therefore, an apropriate classification of GO tools was required. Thus, in order to organize all these resources, an *Standard Functionalities Set* was defined [1]. This list of functionalities allowed to characterize GO tools based on their capabilities. All these information was stored in a database, and it can be accessed by using the query forms of a web aplication called SerbGO. This resource allows to either look for GO tools that fit more or less to the goals that are being pursued, or compare capabilities of different programs. However, many developers still working on GO tools either by improving them or by programming new ones. For this reason, and in order to capture conceptualizations of GO tools capabilities, DeGOT ontology was built. Thus, DeGOT facilitates communication between developers and/or users, as well as the usage of this domain knowledge by computers for multiple purposes.

References

Mosquera, J.L and Sánchez. *SerbGO: searching for the best GO tool.* Nucl. Acids Res. 2008; 36(suppl 2):W368-W371.

Contact

Avinguda Diagonal 645
08028 · Barcelona

Ph. +34 93 402 1560
Fax. +34 93 411 1733
e-mail: jlmosquera@gmail.com

About us

DeGOT ontoloy has been developed by J.L. Mosquera and supervised by A. Sánchez, members of the research group Statistics and Bioinformatics at the Department of Statistics of the University of Barcelona.

GRUP DE RECERCA EN ESTADÍSTICA I BIOINFORMÀTICA

License

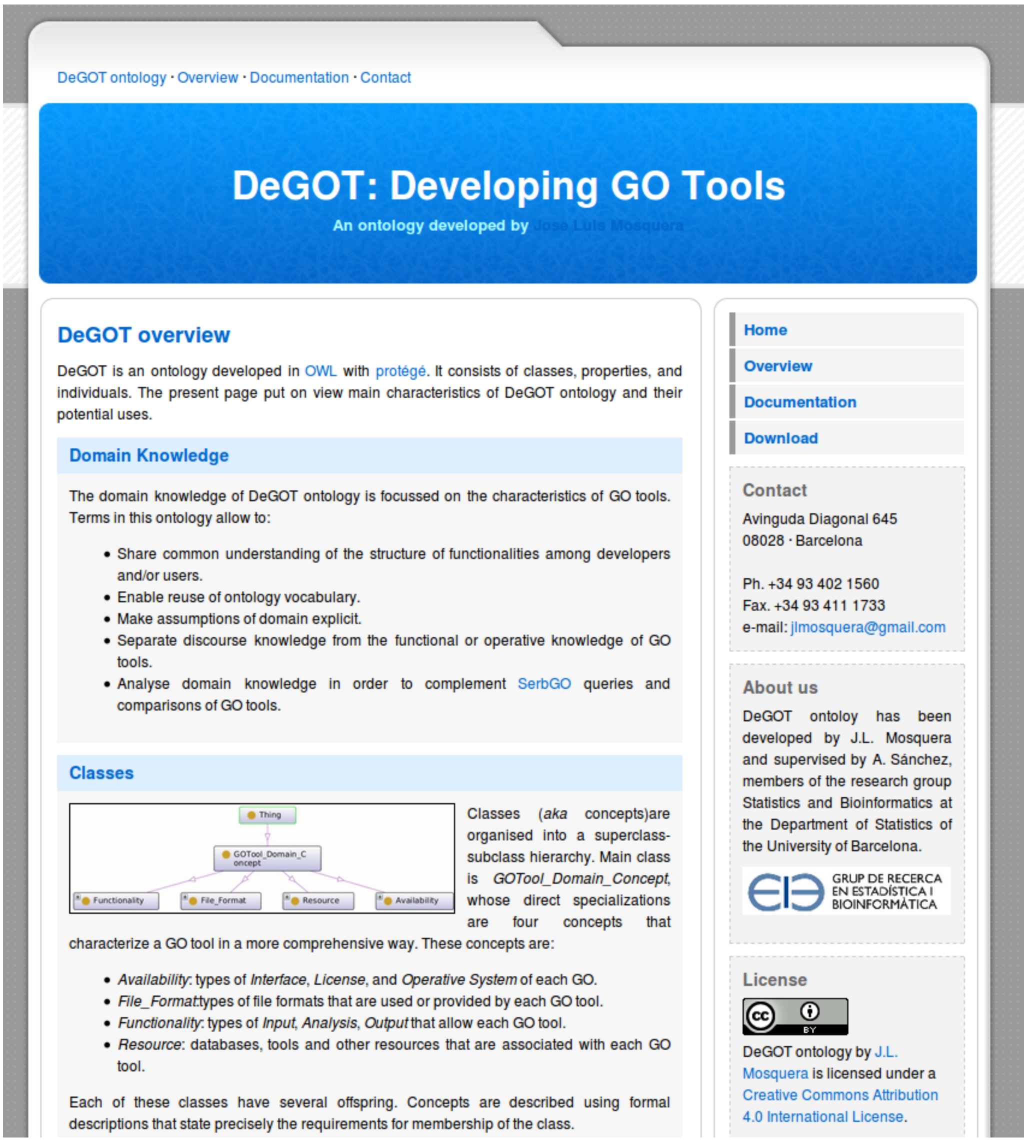
DeGOT ontology by J.L. Mosquera is licensed under a Creative Commons Attribution 4.0 International License.

Figure 7.28: Screenshot of the *Home* page at the DeGOT website.

In order to navigate the ontology, an OWL ([6]) navigator is required, and the best way use is to download and install Protégé software from the website http://protege.stanford.edu/.

7.5.2 General Overview of DeGOT Website

DeGOT website has is structure around four pages. The first page is the *Home* page that welcomes the user (see figure 7.28). The second page is the *Overview* where a general description of the ontology is presented (see figure 7.29).


DeGOT ontology · Overview · Documentation · Contact

DeGOT: Developing GO Tools

An ontology developed by Jose Luis Mosquera

DeGOT overview

DeGOT is an ontology developed in OWL with protégé. It consists of classes, properties, and individuals. The present page put on view main characteristics of DeGOT ontology and their potential uses.

Domain Knowledge

The domain knowledge of DeGOT ontology is focussed on the characteristics of GO tools. Terms in this ontology allow to:

- Share common understanding of the structure of functionalities among developers and/or users.
- Enable reuse of ontology vocabulary.
- Make assumptions of domain explicit.
- Separate discourse knowledge from the functional or operative knowledge of GO tools.
- Analyse domain knowledge in order to complement SerbGO queries and comparisons of GO tools.

Classes

Classes (*aka* concepts)are organised into a superclass-subclass hierarchy. Main class is *GOTool_Domain_Concept*, whose direct specializations are four concepts that characterize a GO tool in a more comprehensive way. These concepts are:

- *Availability*: types of *Interface*, *License*, and *Operative System* of each GO.
- *File_Format*:types of file formats that are used or provided by each GO tool.
- *Functionality*: types of *Input*, *Analysis*, *Output* that allow each GO tool.
- *Resource*: databases, tools and other resources that are associated with each GO tool.

Each of these classes have several offspring. Concepts are described using formal descriptions that state precisely the requirements for membership of the class.

Contact

Avinguda Diagonal 645
08028 · Barcelona

Ph. +34 93 402 1560
Fax. +34 93 411 1733
e-mail: jlmosquera@gmail.com

About us

DeGOT ontoloy has been developed by J.L. Mosquera and supervised by A. Sánchez, members of the research group Statistics and Bioinformatics at the Department of Statistics of the University of Barcelona.

GRUP DE RECERCA EN ESTADÍSTICA I BIOINFORMÀTICA

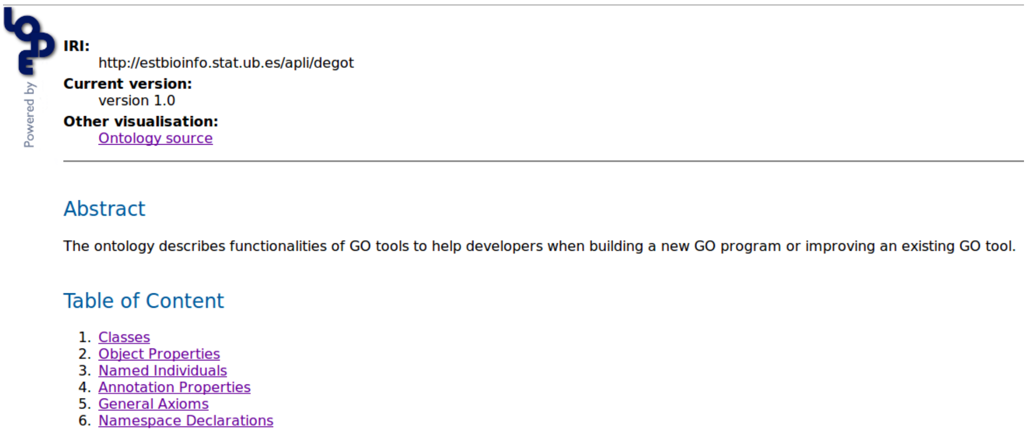
License

DeGOT ontology by J.L. Mosquera is licensed under a Creative Commons Attribution 4.0 International License.

Figure 7.29: Screenshot of the *Overview* page at the DeGOT website.

The third page is the *Documentation* (see figure 7.30). Documentation was obtained by processing the OWL ontology source code through *Live OWL*

Documentation Environment (LODE) ([121]). Here, the user can navigate and read the descriptions and details of each object implemented in DeGOT ontology. Finally, the fourth page is from where the user can *Download* the document tree in OWL code (see figure 7.31).



IRI:
http://estbioinfo.stat.ub.es/apli/degot
Current version:
version 1.0
Other visualisation:
Ontology source

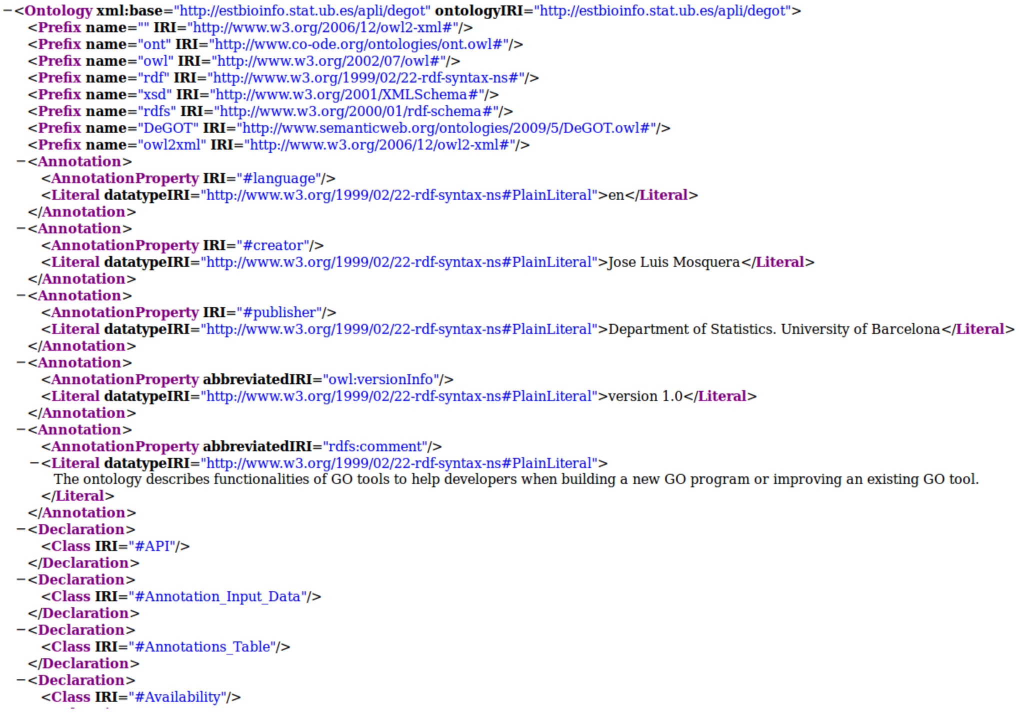
Abstract

The ontology describes functionalities of GO tools to help developers when building a new GO program or improving an existing GO tool.

Table of Content

1. Classes
2. Object Properties
3. Named Individuals
4. Annotation Properties
5. General Axioms
6. Namespace Declarations

Figure 7.30: Screenshot of the *Documentation* page at the DeGOT website.



```
−<Ontology xml:base="http://estbioinfo.stat.ub.es/apli/degot" ontologyIRI="http://estbioinfo.stat.ub.es/apli/degot">
   <Prefix name="" IRI="http://www.w3.org/2006/12/owl2-xml#"/>
   <Prefix name="ont" IRI="http://www.co-ode.org/ontologies/ont.owl#"/>
   <Prefix name="owl" IRI="http://www.w3.org/2002/07/owl#"/>
   <Prefix name="rdf" IRI="http://www.w3.org/1999/02/22-rdf-syntax-ns#"/>
   <Prefix name="xsd" IRI="http://www.w3.org/2001/XMLSchema#"/>
   <Prefix name="rdfs" IRI="http://www.w3.org/2000/01/rdf-schema#"/>
   <Prefix name="DeGOT" IRI="http://www.semanticweb.org/ontologies/2009/5/DeGOT.owl#"/>
   <Prefix name="owl2xml" IRI="http://www.w3.org/2006/12/owl2-xml#"/>
  −<Annotation>
     <AnnotationProperty IRI="#language"/>
     <Literal datatypeIRI="http://www.w3.org/1999/02/22-rdf-syntax-ns#PlainLiteral">en</Literal>
   </Annotation>
  −<Annotation>
     <AnnotationProperty IRI="#creator"/>
     <Literal datatypeIRI="http://www.w3.org/1999/02/22-rdf-syntax-ns#PlainLiteral">Jose Luis Mosquera</Literal>
   </Annotation>
  −<Annotation>
     <AnnotationProperty IRI="#publisher"/>
     <Literal datatypeIRI="http://www.w3.org/1999/02/22-rdf-syntax-ns#PlainLiteral">Department of Statistics. University of Barcelona</Literal>
   </Annotation>
  −<Annotation>
     <AnnotationProperty abbreviatedIRI="owl:versionInfo"/>
     <Literal datatypeIRI="http://www.w3.org/1999/02/22-rdf-syntax-ns#PlainLiteral">version 1.0</Literal>
   </Annotation>
  −<Annotation>
     <AnnotationProperty abbreviatedIRI="rdfs:comment"/>
    −<Literal datatypeIRI="http://www.w3.org/1999/02/22-rdf-syntax-ns#PlainLiteral">
       The ontology describes functionalities of GO tools to help developers when building a new GO program or improving an existing GO tool.
     </Literal>
   </Annotation>
  −<Declaration>
     <Class IRI="#API"/>
   </Declaration>
  −<Declaration>
     <Class IRI="#Annotation_Input_Data"/>
   </Declaration>
  −<Declaration>
     <Class IRI="#Annotations_Table"/>
   </Declaration>
  −<Declaration>
     <Class IRI="#Availability"/>
```

Figure 7.31: Screenshot of the *Download* page at the DeGOT website from where the user can save the documentation tree in OWL code.

The following subsections put show the main characteristics of the ontology and illustrates some examples of their potential uses.

7.5.3 Domain Knowledge of DeGOT Ontology

The domain knowledge of DeGOT ontology is focused on the characteristics of GO tools. Terms in this ontology allow us to:

1. share common understanding of the structure of functionalities among developers and/or users,

2. enable reuse of domain knowledge,

3. make domain assumptions explicit,

4. separate domain knowledge from the functional or operative knowledge of GO tools, and

5. analyze domain knowledge in order to complement SerbGO queries and comparisons of GO tools.

The organization of characteristics into the DeGOT hierarchy makes the upgrading process, including new capabilities, easier than a relational database. The following subsections describe the organization of DeGOT.

7.5.4 Constructs of DeGOT Ontology

DeGOT is an ontology written in OWL. It consists of classes, properties, and individuals. Table 7.24 shows a brief summary of DeGOT ontology metrics.

Object	Num. of Objects
Classes	314
Properties	18
Individuals	4

Table 7.24: Number of objects of DeGOT ontology.

The following paragraphs present what is collected in each of these constructs.

Classes

All ontologies written in `OWL` have a mandatory top class called *Thing*. Classes are organized into a taxonomy or superclass-subclass hierarchy. The subclass of *Thing* is *GOTool_Domain_Concept*. Thus, this class might be considered as the root class of `DeGOT`, whose specializations are four concepts that characterize a GO tool in a more comprehensive way. These concepts are:

- *Availability*: types of *Interface*, *License*, and *Operative System* of the GO tool.
- *File_Format*:types of file formats that are used or provided by each GO tool.
- *Functionality*: types of *Input*, *Analysis*, *Output* that are allowed by the GO tool.
- *Resource*: databases, tools and other resources that are associated with the GO tool.

Figure 7.32 shows the main subclasses of *GOTool_Domain_Concept*.

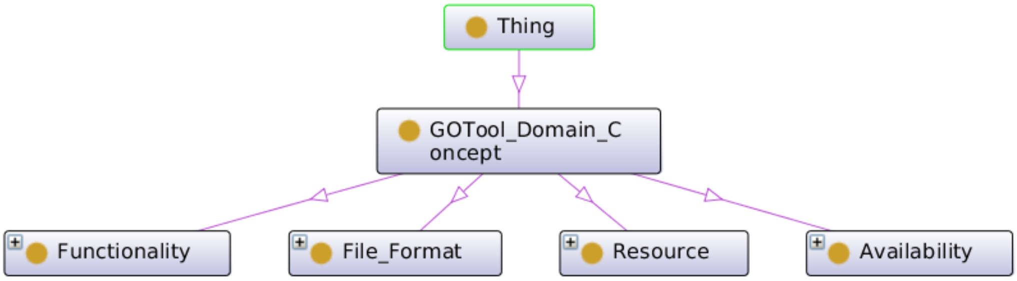
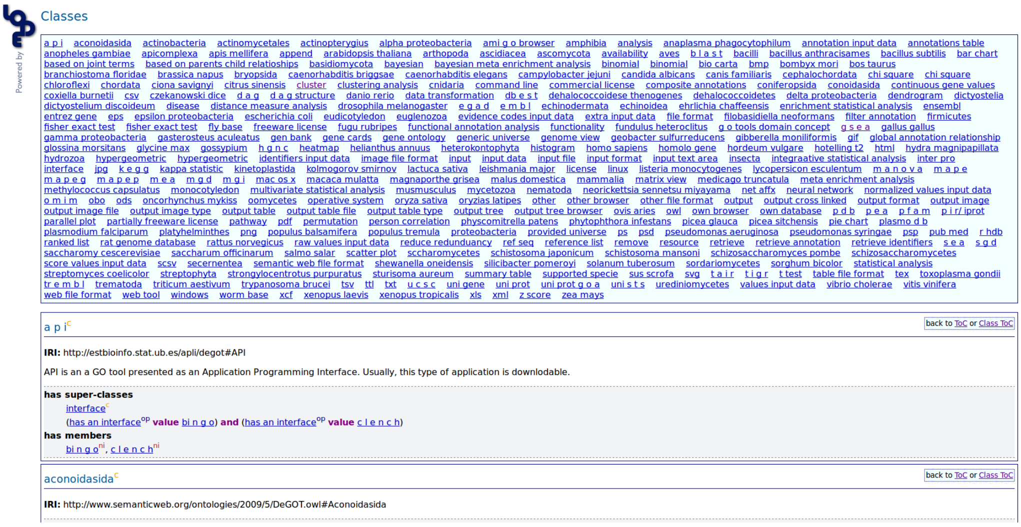


Figure 7.32: DAG of the main classes of `DeGOT` ontology.

These classes have different numbers of offspring. The concepts of each class are detailed in the `DeGOT` documentation, which is available at the website for the ontology (see figure 7.33).



Classes

a p i aconoidasida actinobacteria actinomycetales actinopterygius alpha proteobacteria ami g o browser amphibia analysis anaplasma phagocytophilum annotation input data annotations table
anopheles gambiae apicomplexa apis mellifera append arabidopsis thaliana arthopoda ascidiacea ascomycota availability aves b l a s t bacilli bacillus anthracisames bacillus subtilis bar chart
based on joint terms based on parents child relatioships basidiomycota bayesian bayesian meta enrichment analysis binomial binomial bio carta bmp bombyx mori bos taurus
branchiostoma floridae brassica napus bryopsida caenorhabditis briggsae caenorhabditis elegans campylobacter jejuni candida albicans canis familiaris cephalochordata chi square chi square
chloroflexi chordata ciona savignyi citrus sinensis cluster clustering analysis cnidaria command line commercial license composite annotations coniferopsida conoidasida continuous gene values
coxiella burnetii csv czekanowski dice d a g d a g structure danio rerio data transformation db e s t dehalococcoidese thenogenes dehalococcoidetes delta proteobacteria dendrogram dictyostelia
dictyostelium discoideum disease distance measure analysis drosophila melanogaster e g a d e m b l echinodermata echinoidea ehrlichia chaffeensis enrichment statistical analysis ensembl
entrez gene eps epsilon proteobacteria escherichia coli eudicotyledon euglenozoa evidence codes input data extra input data file format filobasidiella neoformans filter annotation firmicutes
fisher exact test fisher exact test fly base freeware license fugu rubripes functional annotation analysis functionality fundulus heteroclitus g o tools domain concept g s e a gallus gallus
gamma proteobacteria gasterosteus aculeatus gen bank gene cards gene ontology generic universe genome view geobacter sulfurreducens gibberella moniliformis gif global annotation relationship
glossina morsitans glycine max gossypium h g n c heatmap helianthus annuus heterokontophyta histogram homo sapiens homolo gene hordeum vulgare hotelling t2 html hydra magnipapillata
hydrozoa hypergeometric hypergeometric identifiers input data image file format input input data input file input format input text area insecta integraative statistical analysis inter pro
interface jpg k e g g kappa statistic kinetoplastida kolmogorov smirnov lactuca sativa leishmania major license linux listeria monocytogenes lycopersicon esculentum m a n o v a m a p e
m a p e g m a p e p m e a m g d m g i mac os x macaca mulatta magnaporthe grisea malus domestica mammalia matrix view medicago truncatula meta enrichment analysis
methylococcus capsulatus monocotyledon multivariate statistical analysis musmusculus mycetozoa nematoda neorickettsia sennetsu miyayama net affx neural network normalized values input data
o m i m obo ods oncorhynchus mykiss oomycetes operative system oryza sativa oryzias latipes other other browser other file format output output cross linked output format output image
output image file output image type output table output table file output table type output tree output tree browser ovis aries owl own browser own database p d b p e a p f a m p i r/ iprot
parallel plot partially freeware license pathway pdf permutation person correlation physcomitrella patens phytophthora infestans picea glauca picea sitchensis pie chart plasmo d b
plasmodium falciparum platyhelminthes png populus balsamifera populus tremula proteobacteria provided universe ps psd pseudomonas aeruginosa pseudomonas syringae psp pub med r hdb
ranked list rat genome database rattus norvegicus raw values input data reduce redunduancy ref seq reference list remove resource retrieve retrieve annotation retrieve identifiers s e a s g d
saccharomy cescerevisiae saccharum officinarum salmo salar scatter plot sccharomycetes schistosoma japonicum schistosoma mansoni schizosaccharomyces pombe schizosaccharomycetes
score values input data scsv secernentea semantic web file format shewanella oneidensis silicibacter pomeroyi solanum tuberosum sordariomycetes sorghum bicolor statistical analysis
streptomyces coelicolor streptophyta strongylocentrotus purpuratus sturisoma aureum summary table supported specie sus scrofa svg t a i r t i g r t test table file format tex toxoplasma gondii
tr e m b l trematoda triticum aestivum trypanosoma brucei tsv ttl txt u c s c uni gene uni prot uni prot g o a uni s t s urediniomycetes values input data vibrio cholerae vitis vinifera
web file format web tool windows worm base xcf xenopus laevis xenopus tropicalis xls xml z score zea mays

a p i^{c}

back to ToC or Class ToC

IRI: http://estbioinfo.stat.ub.es/apli/degot#API

API is an a GO tool presented as an Application Programming Interface. Usually, this type of application is downloadable.

has super-classes

interface^{c}

(has an interface^{op} **value** bi n g o) **and** (has an interface^{op} **value** c l e n c h)

has members

bi n g o^{ni}, c l e n c h^{ni}

aconoidasida^{c}

back to ToC or Class ToC

IRI: http://www.semanticweb.org/ontologies/2009/5/DeGOT.owl#Aconoidasida

Figure 7.33: Screenshot of the documentation about *Classes* at `DeGOT` website.

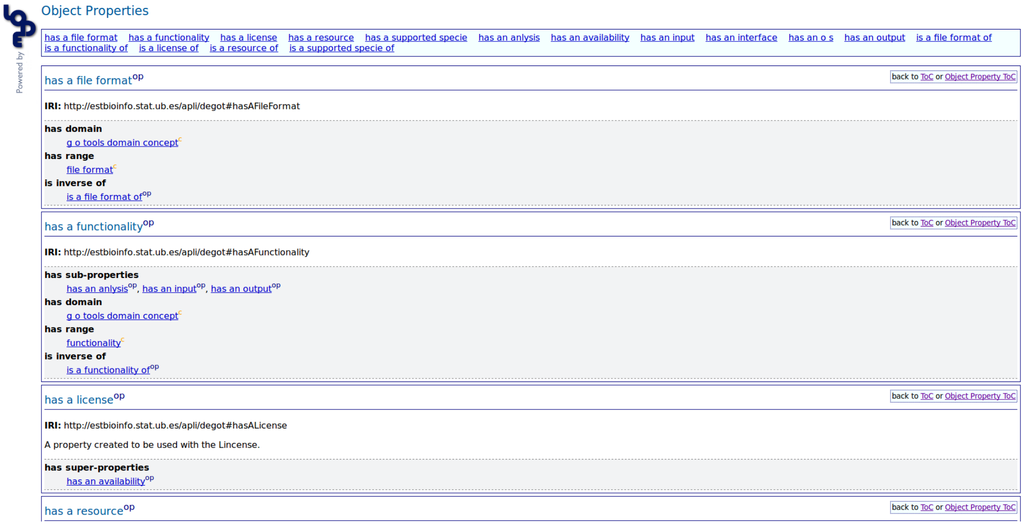
The concepts of the ontology are described using formal descriptions that state precisely the requirements for membership of the class. For example, the concept *Web_Tool* collects all GO tool names that have a web interface.

Classes are divided in two types: subclasses and superclasses. Subclasses are specializations of their superclasses. For instance, consider the classes *Mammalia* and *HomoSapiens*. *HomoSapiens* is a subclass of *Mammalia*, then *Mammalia* is a superclass of the *HomoSapiens* concept. A user can easily understand that,

- "*HomoSapiens* is subsumed by *Mammalia*",
- "All members of the class *HomoSapiens* are members of the class *Mammalia*",
- "All GO tools supporting Homo sapiens species are GO tools supporting Mammalia class",
- ...

Properties

Properties are binary relations of individuals. Strictly speaking, "instances of properties linking individuals". The descriptions of the object properties are available at the website for the ontology (see figure 7.34)



Powered by

Object Properties

has a file format has a functionality has a license has a resource has a supported specie has an anlysis has an availability has an input has an interface has an o s has an output is a file format of is a functionality of is a license of is a resource of is a supported specie of

has a file format^{op}

back to ToC or Object Property ToC

IRI: http://estbioinfo.stat.ub.es/apli/degot#hasAFileFormat

has domain
g o tools domain concept^{c}
has range
file format^{c}
is inverse of
is a file format of^{op}

has a functionality^{op}

back to ToC or Object Property ToC

IRI: http://estbioinfo.stat.ub.es/apli/degot#hasAFunctionality

has sub-properties
has an anlysis^{op}, has an input^{op}, has an output^{op}
has domain
g o tools domain concept^{c}
has range
functionality^{c}
is inverse of
is a functionality of^{op}

has a license^{op}

back to ToC or Object Property ToC

IRI: http://estbioinfo.stat.ub.es/apli/degot#hasALicense

A property created to be used with the Lincense.

has super-properties
has an availability^{op}

has a resource^{op}

back to ToC or Object Property ToC

Figure 7.34: Screenshot of the documentation about *Object Properties* at DeGOT website.

DeGOT ontology has 18 properties. These properties are shown in table 7.25.

Figure 7.35 shows a subgraph of DeGOT ontology where arcs represent different types of links, some of them are properties that have been defined.

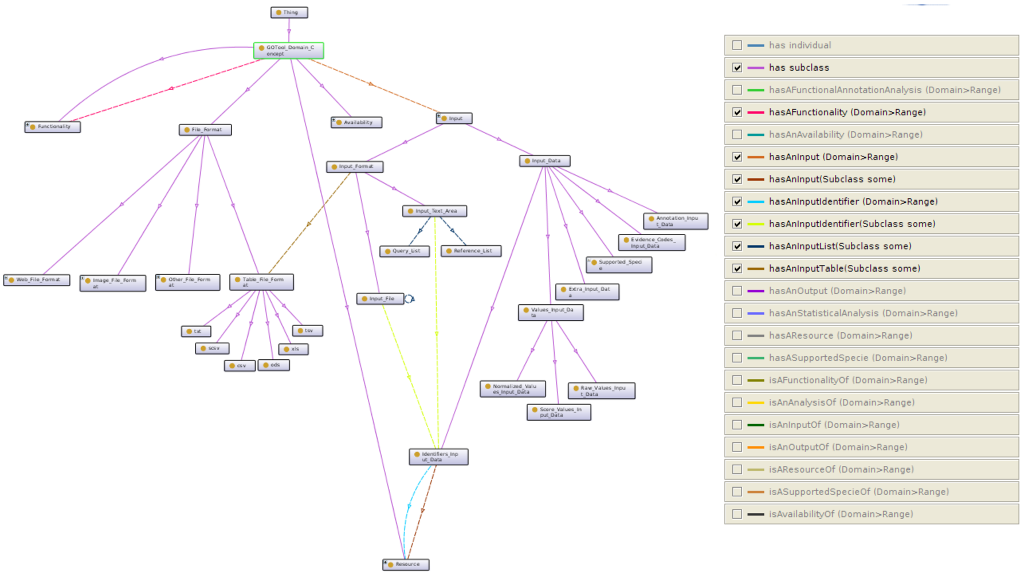
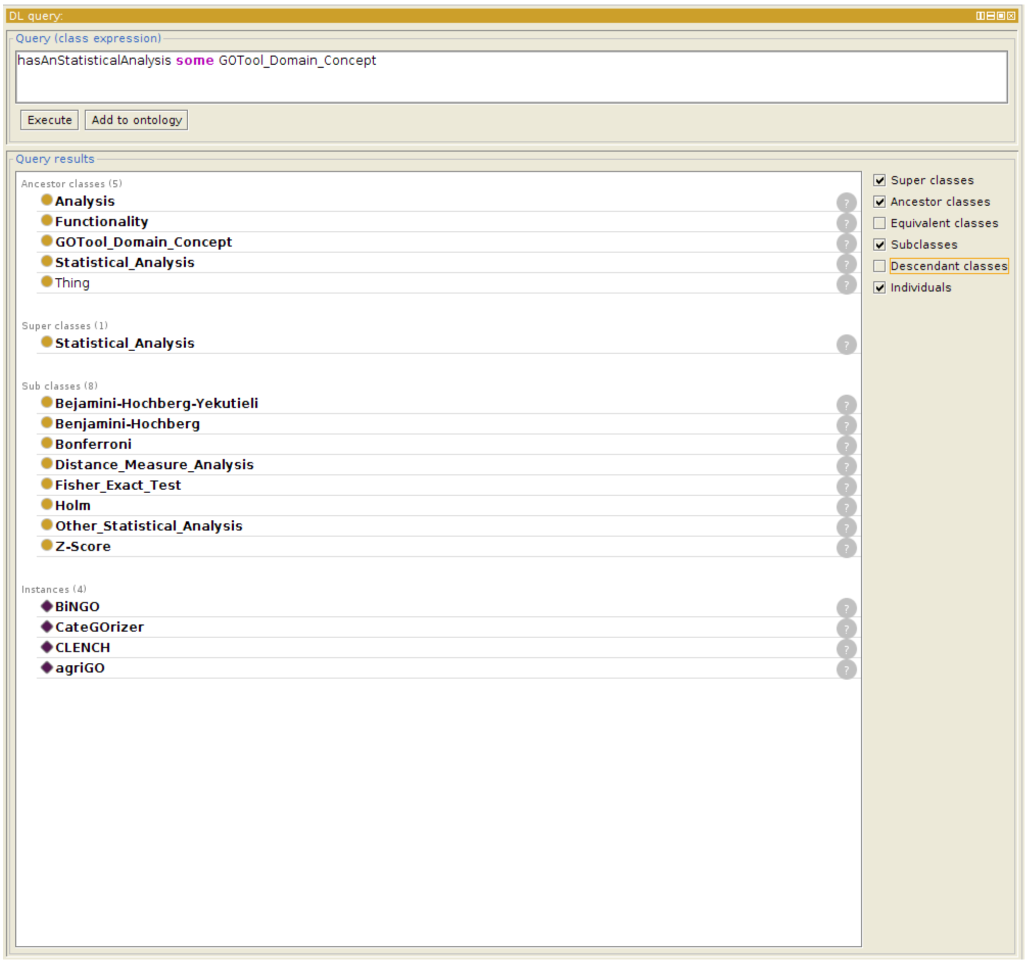

Thing
GOTool_Domain_C oncept
Functionality
File_Format
Availability
Input
Input_Format
Input_Data
Input_Text_Area
Query_List
Reference_List
Annotation_Inpu t_Data
Evidence_Codes_ Input_Data
Supported_Speci e
Extra_Input_Dat a
Values_Input_Da ta
Web_File_Format
Image_File_Form at
Other_File_Form at
Table_File_Form at
Input_File
txt
scsv
csv
ods
xls
tsv
Normalized_Valu es_Input_Data
Raw_Values_Inpu t_Data
Score_Values_In put_Data
Identifiers_Inp ut_Data
Resource
☐ has individual
☑ has subclass
☐ hasAFunctionalAnnotationAnalysis (Domain>Range)
☑ hasAFunctionality (Domain>Range)
☐ hasAnAvailability (Domain>Range)
☑ hasAnInput (Domain>Range)
☑ hasAnInput(Subclass some)
☑ hasAnInputIdentifier (Domain>Range)
☑ hasAnInputIdentifier(Subclass some)
☑ hasAnInputList(Subclass some)
☑ hasAnInputTable(Subclass some)
☐ hasAnOutput (Domain>Range)
☐ hasAnStatisticalAnalysis (Domain>Range)
☐ hasAResource (Domain>Range)
☐ hasASupportedSpecie (Domain>Range)
☐ isAFunctionalityOf (Domain>Range)
☐ isAnAnalysisOf (Domain>Range)
☐ isAnInputOf (Domain>Range)
☐ isAnOutputOf (Domain>Range)
☐ isAResourceOf (Domain>Range)
☐ isASupportedSpecieOf (Domain>Range)
☐ isAvailabilityOf (Domain>Range)

Figure 7.35: Subgraph of DeGOT showing some relations between GO tools assigned to different concepts.

Object Property	Domain	Range	Inverse Property
hasAnAvailability	GOTool_Domain_Concept	Availability	isAvailabilityOf
hasALicense			
hasAnInterface			
hasAnOS			
hasAFunctionality	GOTool_Domain_Concept	Functionality	isFunctionalityOf
hasAnInput	Input	Output	is AnInputOf
hasAnStatisticalAnalysis	Statistical_Analysis	Statistical_Analysis	
hasAFunctionalAnnotationAnalysis	Functional_Annotation_Analysis	Functional_Annotation_Analysis	
hasAnOutput	Output	Output	isAnOutputOf
hasAResource	GOTool_Domain_Concept	Resource	isARecourceOf
hasASupportedSpecie	GOTool_Domain_Concept	Supported_Specie	isASupportedSpecieOf
isAvailabilityOf	Availability	GOTool_Domain_Concept	hasAnAvailability
isAFunctionalityOf	Functionality	GOTool_Domain_Concept	hasAFunctionality
inAnInputOf	Input	GOTool_Domain_Concept	is AnInputOf
isAnAnalysisOf	Analysis	GOTool_Domain_Concept	
isAnOutputOf	Output	GOTool_Domain_Concept	isAnOutputOf
isAResourceOf	Resource	GOTool_Domain_Concept	hasARecource
isASupportedSpecieOf	Supported_Specie	GOTool_Domain_Concept	hasASupportedSpecie

Table 7.25: Properties of DeGOT ontology with the corresponding domains and ranges, and their associated inverses.

Based on the properties of DeGOT, a user may perform different types of queries. For example, suppose that a potential user is interested in looking for different statistical methods, which are available in GO tools annotated in DeGOT, allowing exploration of GO terms. This can be performed by using the property `hasAnStatisticalAnalysis some GOTool_Domain_Concept`. This property allows he/she to link the whole list of GO tool instances to the individuals that perform an exploring analysis of the GO terms.


DL query:

Query (class expression)

hasAnStatisticalAnalysis some GOTool_Domain_Concept

Execute | Add to ontology

Query results

Ancestor classes (5)
- Analysis
- Functionality
- GOTool_Domain_Concept
- Statistical_Analysis
- Thing

Super classes (1)
- Statistical_Analysis

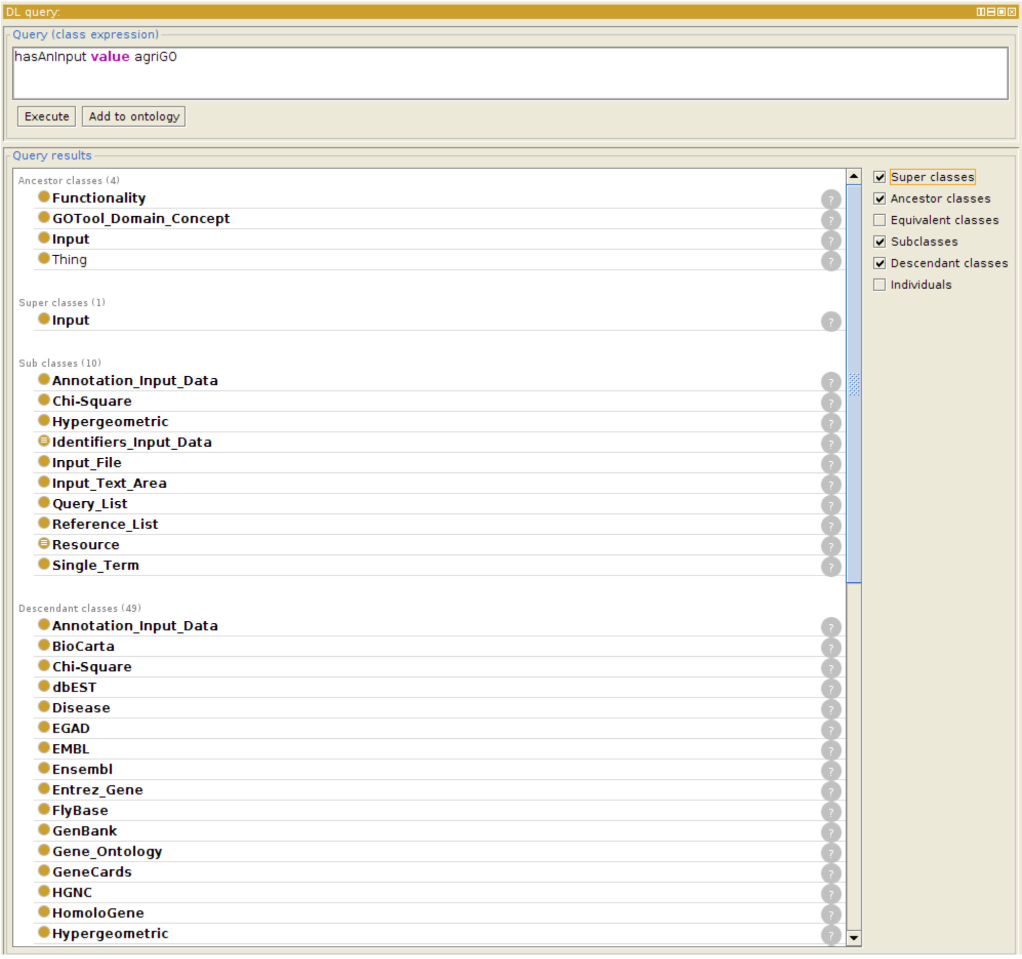
Sub classes (8)
- Bejamini-Hochberg-Yekutieli
- Benjamini-Hochberg
- Bonferroni
- Distance_Measure_Analysis
- Fisher_Exact_Test
- Holm
- Other_Statistical_Analysis
- Z-Score

Instances (4)
- BiNGO
- CateGOrizer
- CLENCH
- agriGO

☑ Super classes
☑ Ancestor classes
☐ Equivalent classes
☑ Subclasses
☐ Descendant classes
☑ Individuals

Figure 7.36: Result of the query `hasAnStatisticalAnalysis some GOTool_Domain_Comcept`.

That is, based on this property the user is asking for GO tools that are annotated in the superclass `Statistical_Analysis`. This class has direct subclasses (i.e. children) and also ancestor classes. Among the direct subclasses there are statistical methods for exploration such as `Distance_Measure_Analysis`, `Fisher_Exact_Test` or `Other_Statistical_Analysis`, and among ancestor classes the user finds `Functionality` or `Analysis`, among others. Figure 7.36 shows the result after querying for this property.



DL query:

Query (class expression)

hasAnInput value agriGO

Query results

Ancestor classes (4)
- Functionality
- GOTool_Domain_Concept
- Input
- Thing

Super classes (1)
- Input

Sub classes (10)
- Annotation_Input_Data
- Chi-Square
- Hypergeometric
- Identifiers_Input_Data
- Input_File
- Input_Text_Area
- Query_List
- Reference_List
- Resource
- Single_Term

Descendant classes (49)
- Annotation_Input_Data
- BioCarta
- Chi-Square
- dbEST
- Disease
- EGAD
- EMBL
- Ensembl
- Entrez_Gene
- FlyBase
- GenBank
- Gene_Ontology
- GeneCards
- HGNC
- HomoloGene
- Hypergeometric

☑ Super classes
☑ Ancestor classes
☐ Equivalent classes
☑ Subclasses
☑ Descendant classes
☐ Individuals

Figure 7.37: Results of the query `hasAnInput value agriGO`.

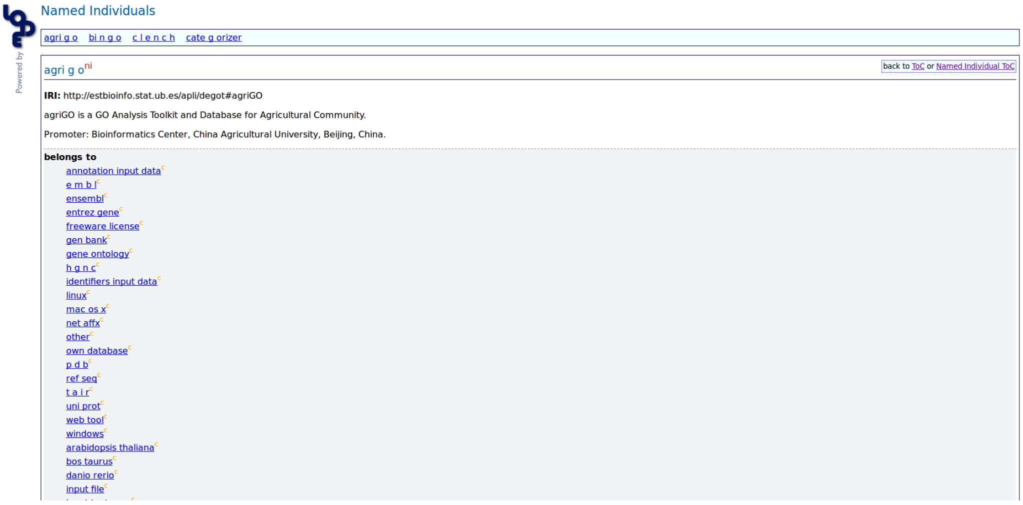
`DeGOT` also provides properties limited to a single value. For instance, imagine that the potential user is now interested in knowing the inputs required by a specific GO tool, for example, `agriGO` ([49]). This question can be answered by applying the property `hasAnInput value agriGO`. What this property does is to query for all *Input* subclasses that hold the GO tool *agriGO*. Figure 7.37 shows an snapshot of the results for such a query.

Note that properties can have inverses (see table 7.25). For instance, the inverse of `hasAnInput` is `isAnInputOf`.

No restrictions have been defined on properties of DeGOT ontology.

Individuals

Individuals annotated in DeGOT classes are the GO tool names. Description and details about the classes where individuals are annotate can be found at the website for the ontology (see figure 7.38).



Named Individuals

agri g o bi n g o c l e n c h cate g orizer

agri g o ni

back to ToC or Named Individual ToC

IRI: http://estbioinfo.stat.ub.es/apli/degot#agriGO

agriGO is a GO Analysis Toolkit and Database for Agricultural Community.

Promoter: Bioinformatics Center, China Agricultural University, Beijing, China.

belongs to

- annotation input data c
- e m b l c
- ensembl c
- entrez gene c
- freeware license c
- gen bank c
- gene ontology c
- h g n c c
- identifiers input data c
- linux c
- mac os x c
- net affx c
- other c
- own database c
- p d b c
- ref seq c
- t a i r c
- uni prot c
- web tool c
- windows c
- arabidopsis thaliana c
- bos taurus c
- danio rerio c
- input file c

Figure 7.38: Screenshot of the documentation about *Named Individuals* at DeGOT website.

Individuals may belong to more than one class. Figure 7.39 shows a representation of some classes of DeGOT ontology containing some individuals.

Some GO tools classified in SerbGO have been annotated in each domain of interest of DeGOT ontology. Each annotated GO tool has been assigned to the respective concepts. That is, each software annotated in DeGOT ontology has been assigned to the most specific classes that define the capabilities of such a GO tool. Figure 7.39 displays a subgraph representation of some classes for agriGO ([49]), BinGO ([99]), CateGOrizer ([78]), and CLENCH ([137]) programs.

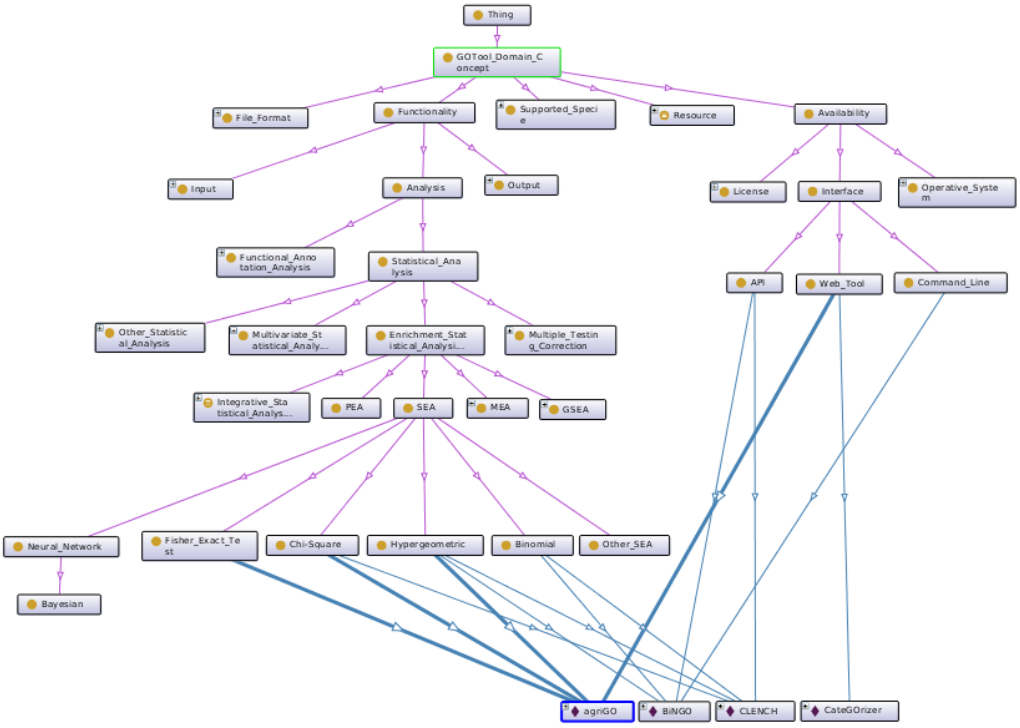


Figure 7.39: Some concepts of `DeGOT` classifying `agriGO`, `BinGO`, `CateGOrizer`, and `CLENCH` tools. Nodes with a diamond show individuals, and nodes with circles show concepts. Magenta arcs stand for a relation of `hasASubclass` and blue arcs indicate which of the most specific categories of the domain are annotating an specific GO tool.

Chapter 8

Discussion

This second part of the thesis is devoted to answer the six specific objectives associated with the study of GO tools for the traditional enrichment analysis (see section 2.2.2.2).

The first contribution of this part was the definition of a Standard Functionalities Set (see section 7.1). It was devoted to classifying the GO tools according to their capabilities. To build this set of functionalities we examined a large list of literature associated with the GO tools for enrichment analysis. The candidate list of GO tools to be examined was extracted from all the GO tools available at the GO Consortium website. After a long period of time and a meticulous study process, we identified all the functionalities that GO tools provided for the different types of analyses. We carried out a standardization process in order to homogenize all the names and features. As a result we proposed a list of 205 characteristics organized into nine main sections of capabilities.

Based on the Standard Functionalities Set we classified a list of 26 GO tools for enrichment analysis that was available at the website of the GO consortium website. The content analysis for performing such a classification consisted in identifying whether the capabilities that each GO tool mentioned in the associated literature reference(s) were actually provided by the software. This task involved executing and trying each GO tool *in situ*, identifying the standard functionalities names in the literature and categorizing each capability as validated, non-validated or missing. This research led us to build several tables showing the *Standard Functionalities Set with* respect to the whole list of names of the GO tools, collecting the availability of each capability for each GO tool and according to the standard functionalities set.

In order to take advantage of such results after evaluating and classifying the list of selected GO tools, we developed a web-based application, called `SerbGO` ([111]) (see section r:serbgo). It is devoted to helping users with the selection and comparison of software for dealing with enrichment analysis that best suits their objectives. SerbGO consists of a database that collects all the information from the tables mentioned above, and two forms to access the information stored in the database. The first form consists of six pages that allow the user to select a list of desired capacities from the Standard Functionalities Set, and after running the tool to obtain the GO tools that satisfy such capabilities. The second form was only one page that contained the whole list of GO tools. This form allows the user to select the GO tools that he/she would like to compare. After running the program, SerbGO yields a large table with all the features that each selected GO tool provide. Nowadays, the SerbGO database stores the information associated with the classification of a list of 50 GO tools.

All the GO tools stored in SerbGO have been monitored over time. This process allowed us to detect that developers introduced improvements and new capabilities in their software. Therefore, we wondered whether may be the different applications might be clustered according to their "specializations". In this regard, we decided to perform a statistical analysis devoted to understand the evolution of GO tools and identifying, if so, the existence of some representative models of GO tools (see section 7.4). This statistical analysis consisted of three parts: descriptive statistics, inferential analysis and multivariate analysis.

Data analysis was focused on information from the 26 original GO tools stored in SerbGO database tables. We considered the tables at three transversal points over time: 2005, 2007 and 2009. These tables required a homogenization process intended to eliminate "redundancies" in the original 205 standard functionalities. This process left us with 178 independent functionalities (see section 6.4.1.2).

Descriptive statistics were divided in two parts; a global analysis and an analysis by sections of functionalities. Global descriptive statistics showed some differences between years in terms of distributions of frequencies. We observed a considerable increase in functionalities between 2005 and 2007, and also between 2007 and 2009. But, in this second period these differences were fewer. These facts suggested us that developers might be investing more effort during the first period of time than during the second. Descriptive statistics by sections suggested again that all the sections of functionalities experienced changes. These improvements were especially notable in functionalities of supported species and annotations. However, improvements in the functionalities of the type of tool, statistical methods and outputs seemed more “homogeneous”. With regard to the input data, frequencies did not suggest great changes in functionalities.

Inferential analysis was also divided into a global analysis and an analysis by sections of functionalities. In global inferential Chi-squared tests of homogeneity showed that there were statistically significant differences between the distribution of frequencies between 2005 and 2007, and between 2005 and 2009. Boxplots and scatterplots with Loess curves and their confidence bands highlighted the overall upward trend whatever the combination of years considered in the plots. Inferential analysis by sections did not show statistically significant differences in most of the comparisons. But, there were statistically significant differences in supported species between the distributions of frequencies in the comparisons between 2005 and 2007, and between 2005 and 2009, as well as between all the comparisons in the annotations section. The overall upward trend in all the boxplots and scatterplots also became evident. However, in sections of statistical methods and outputs it was harder to appreciate such differences between the years 2007 and 2009.

Multivariate analysis consisted in the selection process of the number of clusters, the hierarchical cluster analyses, the multidimensional scaling (MDS), and the Mantel tests. Generally speaking, in all cases we detected a clear lack of substantial cluster structures. Depending on the year, the number of clusters appeared to be low or high. This made us think that it might be associated with the improvements introduced by GO tools developers during the period from 2005 to 2007 in order to fill a “gap” in capabilities offered by the programs. Whatever the year, the measure of similarity used, and leaving aside any limitations due to low Silhouette Coefficient (SC) values, bar diagrams of the average silhouette widths for each number of clusters revealed that partitioning the space with a low number of clusters seemed to explain a substantial part of the information about the groups of GO tools. Supported by this fact we considered three main clusters that could be used to understand the evolution of the GO tools. In hierarchical cluster analysis, regardless of both the year and the measure of similarity used, by cutting dendrograms at the appropriate distance to get the three predetermined clusters, the plots revealed the existence of a “major” cluster that grouped together most of the GO tools, and two “minor” clusters containing one or very few tools. However, when we distinguished the plots by both measures of similarity and years, this finding had to be qualified. In the case of the Jaccard coefficient, we observed that, as time went on, the “major” cluster drew together practically every GO tool, leaving only one tool in each of the “minor” clusters. But, in the case of the Matching coefficient, as time went on “minor” clusters became more prominent and took some tools from the “major” cluster. Therefore, these results pointed us towards the idea of the presence of a certain homogeneity among GO tools over time, when we used the Jaccard coefficient, which was in consonance with the lack of cluster structures suggested by the SCs, and an evolution of GO tools from the homogeneity of capabilities to some sort of “specialization” of programs over time, when we used the Matching coefficient. Generally speaking, MDS results suggested to us that GO tools were more spread out in MDS solutions based on the Jaccard coefficient than in MDS solutions based on the Matching coefficient, regardless of the approach used. We detected some subtle differences in “meaning”, but in general, the behavior of the distances among points in two-dimensional representations were similar with the exception of non-metric MDS solution based on a Matching coefficient in 2009. Clusters of GO tools determined with silhouettes and identified in dendrograms did not show any outstanding separation effect among them, in classical MDS solution based on the Jaccard coefficients. In contrast, we observed that, based on the Matching coefficient, both classical and non-metric approaches seemed to place the clusters of GO tools in the same areas. We highlighted that this assertion was not free of criticism. However, we pointed out that in classical MDS solutions based on the Matching coefficient the clusters were clearly less fuzzy and the GO tools were plotted much closer to each other than in the classical MDS solutions based on the Jaccard coefficient. Non-metric MDS solutions behaved similarly to classical MDS solutions in 2005 and 2007. However, this behavior was completely different in 2009. We observed a contraction of points in 2007 and an expansion in 2009. This fact had no easy interpretation, even more so when considering the poor goodness of fit suggested by the stress values. In our opinion, this fact was a consequence of the evolution and specialization of GO tools over time, which we already highlighted in descriptive statistics and inferential analysis, but which was not in turn clear enough to reflect the changes in terms of distances and clusters. The adequacy plots associated with classical MDS solutions did not suggest bad representations for the two-dimensional configuration. However, the Scree plots and Shepard diagrams associated with non-metric MDS solutions showed *a priori* some discrepancies. On the one hand, the Scree plots showed fair or poor goodnesses of fit for the two-dimensional configuration and suggested that for achieving “optimal” representations, the number of dimensions should be closer to 7. On the other hand, the Shepard diagrams suggested that distances and disparities were good when approximating the proximities between GO tools. Hence, a graphical identification of clusters of GO tools was apparently observed. However, the points in a Shepard diagram are not geometric projections of the proximities, because this type of plot projects a dissimilarity matrix to fewer dimensions. In consequence, we concluded that potential models or specializations of groups of GO tools observed a priori in non-metric MDS solutions were not believable.

We performed two types of Mantel Tests in order to study the relationships between the dissimilarity matrices of the GO tools over time: Simple Mantel Tests and Partial Mantel Tests!Partial Mantel Test. Simple Mantel Tests showed that regardless of the coefficient of similarity used some sorts of relationships between each pair of dissimilarity matrices of GO tools were found. However, the associated coefficients of correlation were poor with the exception of the comparison between dissimilarity matrices for the years 2007 and 2009. This fact suggested to us that the similarities between GO tools evolve over time in the same way in one respect. Nonetheless, Partial Mantel Tests indicated, regardless once again of the coefficient of similarity used, that the matrix of dissimilarities between each pair of GO tools in 2005 did not show a linear relationship with the matrix of dissimilarities in 2009 because they took into account the matrix of dissimilarities in 2007. That brought us to the next conclusion, that the dissimilarities between GO tools did not seem to be the “same” during this period of time, because the third matrix in action removed some type of spurious effect of correlation. Thus, based on the results of the Mantel Tests we concluded that the dissimilarities between each pair of GO tools evolved independently over time.

After surveying the results of the statistical analysis for studying the evolution of GO tools, and by observing the GO Consortium website, we detected that the scientific community not only improved old GO tools, but also developed applications for the enrichment analysis based on new methods and approaches (e.g. `BiNGO` ([99]), `STRING` ([57]), `Blast2GO` ([34]), etc.). In other words, continuous development and improvement of methods to deal with functional annotation has stimulated a considerable increase in the development of GO tools for enrichment analysis. Thus, we decided to develop an ontology devoted to providing developers with a vocabulary that helps them to design new GO tools, as well as to be used as complementary software for `SerbGO`. That is, `DeGOT` (see section 7.5) is an ontology focused on the characteristics of GO tools, whose terms allow us to share common understanding of the structure of functionalities among developers and/or users, in order to enable reuse of domain knowledge, to make domain assumptions explicit, to separate domain knowledge from the functional or operative knowledge of GO tools, and also to analyze domain knowledge in order to complement `SerbGO` queries and comparisons of GO tools. At the end of this thesis, `DeGOT` consisted of 314 classes, 16 object properties and 4 named individuals, which may be easily extended.

During the last years the growth of integrative studies with omics data and/or combination of methods in order to improve the biological knowledge has experimented a remarkable upwards change. In this regard, one of the most interesting extensions that could be carried out in a future line of research should be to investigate the possibility of combining different methods and tools for enrichment analysis in order to provide more plausible and informative results. Moreover, although the classification process and subsequent monitoring is a task of a considerable magnitude in terms of resources and time, it would be interesting to build a tool like `SerbGO`, that it would enable us to classify the new fashion of tools for enrichment analysis which is based on the biological network analysis.

Part IV

Conclusions

Conclusions

This thesis has focused on methods and tools for assigning biological interpretation based on the *Gene Ontology* to data generated in omics experiments. The research has explored two main aspects:

1. The study of two types of semantic similarity measures for exploring GO categories.

2. The classification and study of GO Tools for enrichment analysis.

- With regard to the first issue:

    1. It has been proved that:

        (a) The accessibility matrix associated with a symmetric graph, is symmetric.

        (b) The *Handshaking Theorem* and its corollary can be demonstrated based on the incidence matrix.

        (c) The monotonic property of the probability can be verified in terms of Carey's framework.

        (d) The root node of an ontology is the term with the lowest information content, which in fact is null.

        (e) In order to compute the *Information Content*, the product of the matrix with the number of paths of any length between each pair of terms by the mapping matrix can be used for computing the number of times that each term or any of its specializations appear in the ontology.

        (f) The second Resnik's measure redefined in terms of distance, is a metric distance.

        (g) When restricting to comparable terms, the pseudo-distance of the minimum chain length is a metric distance.

    2. It has been shown that:

        (a) There exists a certain level of analogy between the *Object-Ontology Complex* concept and *Partially Ordered Sets Ontology.*

        (b) The Lord's measure is in fact the Resnik's measure.

3. It has been developed an `R` package called `sims` that:
    (a) It is addressed for computing semantic similarity.
    (b) It has implemented a large number of measures from two different approaches.
    (c) It provides an alternative point of view for comparing two lists of genes based on semantic similarity profiles.
    (d) It is freely available at the `GitHub` repository `https://github.com/jlmosquera/sims`.

- With regard to the second issue:

    1. It has been shown that the definition of an *Standard Functionalities Set* allows us to classify GO tools for enrichment analysis.
    2. It has been developed a web-based tool called `SerbGO` that:
        (a) It is addressed for selecting and comparing GO tools for enrichment analysis.
        (b) It is freely available at the server of the *Statistics and Bioinformatics Research Group* (`http://estbioinfo.stat.ub.es/apli/serbgo`).
    3. The study of the GO tools has revealed that:
        (a) Promoters have been introducing improvements on the GO tools for enrichment analysis over time.
        (b) GO tools evolved homogeneously and no clears groups of GO tools has been found.
    4. It has been developed an ontology called `DeGOT` that:
        (a) It provides an organized vocabulary for helping developers when they need either to design a new GO tool or to improve an existing one.
        (b) It can be used for supporting queries and comparisons of GO tools performed with `SerbGO`.
        (c) It is freely available at the server of the *Statistics and Bioinformatics Research Group* (`http://estbioinfo.stat.ub.es/apli/degot`).

Bibliography

[1] Mehdi Achour, Friedhelm Betz, Antony Dovgal, Nuno Lopes, Hannes Magnusson, Georg Richter, Damien Seguy, and Jakub Vrana. *PHP Manual*. PHP Documentation Group, olson, philip edition, Mar 2014.

[2] Daniel Adler, Duncan Murdoch, and others. *rgl: 3D visualization device system (OpenGL)*, 2014. R package version 0.93.1098.

[3] Fátima Al-Shahrour, Ramón Díaz-Uriarte, and Joaquín Dopazo. FatiGO: a web tool for finding significant associations of gene ontology terms with groups of genes. *Bioinformatics*, 20(4):578–580, 2004.

[4] Gil Alterovitz and Marco Ramoni, editors. *Knowledge-Based Bioinformatics. From Analysis to Interpretation.* John Wiley and Sons, Ltd., Publication, 2010.

[5] W3C Archives. Resource Description Framework (RDF). http://www.w3.org/RDF/.

[6] W3C Archives. Web Ontology Language (OWL). http://www.w3.org/OWL/.

[7] W3C Archives. XML technology. http://www.w3.org/standards/xml/.

[8] F. Gregory Ashby and Daniel M. Ennis. Similarity measures. *Scholarpedia*, (12):4116, 2007.

[9] Several authors. Gene Ontology Tools. http://neurolex.org/wiki/Category:Resource:Gene_Ontology_-Tools#tab=Basic.

[10] Kenneth Baclawski and Tianhua Niu. *Ontologies for Bioinformatics.* Computational molecular biology. MIT Press, 2006.

[11] Douglas Bates and Martin Maechler. *Matrix: Sparse and Dense Matrix Classes and Methods*, 2014. R package version 1.1-4.

[12] Tim Beißbarth and Terence P. Speed. GOstat: find statistically overrepresented Gene Ontologies within a group of genes. *Bioinformatics*, 20(9):1464–1465, 2004.

[13] Vidar Beisvag, Frode Junge, Hallgeir Bergum, Lars Jolsum, Stian Lydersen, Clara-Cecilie Gunther, Heri Ramampiaro, Mette Langaas, Arne Sandvik, and Astrid Laegreid. GeneTools - application for functional annotation and statistical hypothesis testing. *BMC Bioinformatics*, 7(1):470, 2006.

[14] Tim Berners-Lee, James Hendler, and Ora Lassila. The semantic web. *Scientific American Magazine*, 2001.

[15] Abraham Bernstein, Esther Kaufmann, Christoph Bürki, and Mark Klein. How similar is it? towards personalized measures in ontologies. In *7 Internationale Tagung Wirtschaftsinformatik*, 2005. Bamberg, Germany.

[16] Gabriel F. Berriz, Oliver D. King, Barbara Bryant, Chris Sander, and Frederick P. Roth. Characterizing gene sets with FuncAssociate. *Bioinformatics*, 19(18):2502–2504, 2003.

[17] John Adrian Bondy and Uppaluri S. R. Murty. *Graph Theory with Applications*. North Holland, 1976.

[18] Ingwer Borg and Patrick J. F. Groenen. *Modern Multidimensional Scaling: Theory and Applications*. Springer Series in Statistics. Springer, 2005.

[19] John Boyle. SeqExpress: desktop analysis and visualization tool for gene expression experiments. *Bioinformatics*, 20(10):1649–1650, 2004.

[20] Andreas Buja, Deborah F. Wayne, Michael L. Littman, Nathaniel Dean, Heike Hofmann, and Lisha Chen. Data visualization with multidimensional scaling. *Journal of Computational and Graphical Statistics*, 17(2):444–472, 2008.

[21] Kimberly Bussey, David Kane, Margot Sunshine, Sudar Narasimhan, Satoshi Nishizuka, William Reinhold, Barry Zeeberg, Weinstein Ajay, and John N. Weinstein. MatchMiner: a tool for batch navigation among gene and gene product identifiers. *Genome Biology*, 4(4):R27, 2003.

[22] Seth Carbon, Amelia Ireland, Christopher J. Mungall, ShenQiang Shu, Brad Marshall, Suzanna Lewis, the AmiGO Hub, and the Web Presence Working Group. AmiGO: online access to ontology and annotation data. *Bioinformatics*, 25(2):288–289, 2009.

[23] Vincent J. Carey. Ontology concepts and tools for statistical genomics. *Journal of Multivariate Analysis*, 90:213–228, 2003.

[24] P. Carmona-Saez, M. Chagoyen, F. Tirado, J. M. Carazo, and A. Pascual-Montano. GENECODIS: a web-based tool for finding significant concurrent annotations in gene lists. *Genome Biol.*, 8(1):R3, 2007.

[25] Manuel Castellet and Irene Llerena. *Àlgebra lineal i geometria.* Universitat Autónoma de Barcelona, 2nd edition, 1990.

[26] Cristian I. Castillo-Davis and Daniel L. Hartl. GeneMerge—post-genomic analysis, data mining, and hypothesis testing. *Bioinformatics*, 19(7):891–892, 2003.

[27] Tzu-Chiao Chao and Nicole Hansmeier. Microfluidic devices for high-throughput proteome analyses. *Proteomics*, 13(3-4):467–479, 2013.

[28] Xiangsheng Chen, Jiuyong Li, Grant Daggard, and Xiaodi Huang. Finding similar patterns in microarray data. In Shichao Zhang and Ray Jarvis, editors, *AI 2005: Advances in Artificial Intelligence*, volume 3809 of *Lecture Notes in Computer Science*, pages 1272–1276. Springer Berlin Heidelberg, 2005.

[29] Jill Cheng, Shaw Sun, Adam Tracy, Earl Hubbell, Joseph Morris, Venu Valmeekam, Andrew Kimbrough, Melissa S. Cline, Guoying Liu, Ron Shigeta, David Kulp, and Michael A. Siani-Rose. NetAffx Gene Ontology Mining Tool: a visual approach for microarray data analysis. *Bioinformatics*, 20(9):1462–1463, 2004.

[30] William S. Cleveland. Robust locally weighted regression and smoothing scatterplots. *Journal of the American Statistical Association*, 74(368):829–836, 1979.

[31] William S. Cleveland and Susan J. Devlin. Locally weighted regression: An approach to regression analysis by local fitting. *Journal of the American Statistical Association*, 83(403):596–610, 1988.

[32] William S. Cleveland, Eric Grosse, and Ming Jen Shyu. Local regression models. In John M. Chambers and Trevor Hastie, editors, *Statistical Models in S*, pages 309–376. Chapman & Hall, New York., 1992.

[33] William S. Cleveland and Clive L. Loader. Smoothing by local regression: Principles and methods. In Wolfang Häerdle and Michael

G. Schimek, editors, *Statistical Theory and Computational Aspects of Smoothing*, Contributions to Statistics, pages 10–49. Physica-Verlag Heidelberg, 1996.

[34] Ana Conesa, Stefan Götz, Juan Manuel García-Gómez, Javier Terol, Manuel Talón, and Montserrat Robles. Blast2GO: a universal tool for annotation, visualization and analysis in functional genomics research. *Bioinformatics*, 21(18):3674–3676, 2005.

[35] The UniProt Consortium. Activities at the Universal Protein Resource (UniProt). *Nucleic Acids Res.*, 42(Database issue):D191–198, 2014.

[36] Trevor F. Cox and Michael A. A. Cox. *Multidimensional Scaling, Second Edition*. Chapman & Hall/CRC Monographs on Statistics & Applied Probability. Taylor & Francis, second edition edition, 2010.

[37] Gabor Csardi and Tamas Nepusz. The igraph software package for complex network research. *InterJournal*, Complex Systems:1695, 2006.

[38] Carles M. Cuadras. Distancias estadísticas. *Estadística Española*, 30(119):295–378, 1989.

[39] B.A. Davey and H.A. Priestley. *Introduction to Lattices and Order*. Cambridge University Press, 2nd edition, 1990.

[40] Glynn Dennis, Brad Sherman, Douglas Hosack, Jun Yang, Wei Gao, H. Clifford Lane, and Richard Lempicki. David: Database for annotation, visualization, and integrated discovery. *Genome Biology*, 4(5):P3, 2003.

[41] Ramón Díaz-Uriarte, Fátima Al-Shahrour, and Joaquín Dopazo. The use of go terms to understand the biological significance of microarray differential gene expression data. In Kimberly K. Johnson and Simon M. Lin, editors, *Methods of Microarray Data Analysis III*, pages 233–248. Kluwer Academic Publishers, 2003.

[42] Maximilian Diehn, Gavin Sherlock, Gail Binkley, Heng Jin, John C. Matese, Tina Hernandez-Boussard, Christian A. Rees, J. Michael Cherry, David Bolstein, Patrick O. Brown, and Ash A. Alizadeth. SOURCE: a unified genomic resource of functional annotations, ontologies, and gene expression data. *Nucleic Acids Research*, 31(1):219–223, 2003.

[43] Reinhard Diestel. *Graph Theory*, volume 173 of *Graduate Texts in Mathematics*. Springer, 4th edition, 2010.

[44] Scott W. Doniger, Nathan Salomonis, Kam D. Dahlquist, Karen Vranizan, Steven C. Lawlor, and Bruce R. Conklin. MAPPFinder: using Gene Ontology and GenMAPP to create a global gene-expression profile from microarray data. *Genome Biology*, 4(R7), 2003.

[45] Joaquín Dopazo. Functional interpretation of microarray experiments. *OMICS*, 10(3):398–410, 2006.

[46] Sorin Draghici, Purvesh Khatri, Pratik Bhavsar, Abhik Shah, Stephen A. Krawetz, and Michael A. Tainsky. Onto-Tools, the toolkit of the modern biologist: Onto-Express, Onto-Compare, Onto-Design and Onto-Translate. *Nucleic Acids Research*, 31(13):3775–3781, 2003.

[47] Sorin Drăghici, Purvesh Khatri, Rui P. Martins, G. Charles Ostermeier, and Stephen A. Krawetz. Global functional profiling of gene expression. *Genomics*, 81(2):98–104, 2003.

[48] Zhidian Du, Lin Li, Chin-Fu Chen, Philip S. Yu, and James Z. Wang. G-SESAME: web tools for go-term-based gene similarity analysis and knowledge discovery. *Nucleic Acids Research*, 37(Web Server Issue):W345–W349, 2009.

[49] Zhou Du, Xin Zhou, Yi Ling, Zhenhai Zhang, and Zhen Su. agriGO: a GO analysis toolkit for the agricultural community. *Nucleic Acids Research*, 38(Supp.):W64–W70, 2010.

[50] Karen Eilbeck, Suzanna E. Lewis, Christopher J. Mungall, Mark Yandell, Lincoln Stein, Richard Durbin, and Michael Ashburner. The sequence ontology: A tool for the unification of genome annotations. *Genome Biology*, 6(5):R44, 2005.

[51] Herbert B. Enderton. *Elements of Set Theory.* Academis Press, Inc., 1977.

[52] Brian S. Everitt. *An R and S-Plus® Companion to Multivariate Analysis.* Springer Texts in Statistics. Springer, 2006.

[53] Brian S. Everitt, Sabine Landau, Morven Leese, and Daniel Stahl. *Cluster Analysis.* Wiley Series in Probability and Statistics. John Wiley & Sons, Ltd, 5th edition, 2011.

[54] Seth Falcon and Robert Gentleman. Using GOstats to test gene lists for GO term association. *Bioinformatics*, 23(2):257–258, 2007.

[55] William Feller. *An Introduction to Probability Theory and Its Applications*, volume 1 of *Wiley Series in Probability and Methematical Statistics*. John Wiley & Sons, Inc., 3rd edition, 1968.

[56] Paul Flicek, M. Ridwan Amode, Daniel Barrell, Kathryn Beal, Konstantinos Billis, Simon Brent, Denise Carvalho-Silva, Peter Clapham, Guy Coates, Stephen Fitzgerald, Laurent Gil, Carlos García Girón, Leo Gordon, Thibaut Hourlier, Sarah Hunt, Nathan Johnson, Thomas Juettemann, Andreas K. Kähäri, Stephen Keenan, Eugene Kulesha, Fergal J. Martin, Thomas Maurel, William M. McLaren, Daniel N. Murphy, Rishi Nag, Bert Overduin, Miguel Pignatelli, Bethan Pritchard, Emily Pritchard, Harpreet S. Riat, Magali Ruffier, Daniel Sheppard, Kieron Taylor, Anja Thormann, Stephen J. Trevanion, Alessandro Vullo, Steven P. Wilder, Mark Wilson, Amonida Zadissa, Bronwen L. Aken, Ewan Birney, Fiona Cunningham, Jennifer Harrow, Javier Herrero, Tim J.P. Hubbard, Rhoda Kinsella, Matthieu Muffato, Anne Parker, Giulietta Spudich, Andy Yates, Daniel R. Zerbino, and Stephen M.J. Searle. Ensembl 2014. *Nucleic Acids Res.*, 42(Database issue):D749–755, 2014.

[57] Andrea Franceschini, Damina Szklarczyk, Sune Frankild, Michael Kuhn, Milan Simonovic, Alexander Roth, Jianyi Lin, Pablo Minguez, Peer Bork, Christian von Mering, and Lars J. Jensen. STRING v9.1: protein-protein interaction networks, with increased coverage and integration. *Nucleic Acids Res.*, 41(Database issue):D808–815, 2013.

[58] Caroline C. Friedel. *Bioinformatics methods for the biological interpretation of high-throughput experiments.* PhD thesis, Universitt München, 2014.

[59] Holger Froehlich. GOSim - An R-Package for Computation of Information Theoretic GO Similarities Between Terms and Gene Products. *BMC Bioinformatics*, (8):166, 2007.

[60] Mingxin Gan, Xue Dou, and Rui Jiang. From ontology to semantic similarity: Calculation of ontology-based semantic similarity. *The Scientific World Journal*, (793091):1–11, 2013.

[61] GeneSifter©. http://www.genesifter.net/web/.

[62] Robert C. Gentleman, Vincent J. Carey, Douglas M. Bates, Ben Bolstad, Marcel Dettling, Sandrine Dudoit, Byron Ellis, Laurent Gautier, Yongchao Ge, Jeff Gentry, Kurt Hornik, Torsten Hothorn, Wolfgang

Huber, Stefano Iacus, Rafael Irizarry, Friedrich Leisch, Cheng Li, Martin Maechler, Anthony J. Rossini, Gunther Sawitzki, Colin Smith, Gordon Smyth, Luke Tierney, Jean Y.H. Yang, and Jianhua Zhang. Bioconductor: open software development for computational biology and bioinformatics. *Genome Biology*, 5(10):R80, 2004.

[63] Robert C. Gentleman, Vincent J. Carey, Douglas M. Bates, and others. Bioconductor: Open software development for computational biology and bioinformatics. *Genome Biology*, 5:R80, 2004.

[64] Vincent Goulet, Christophe Dutang, Martin Maechler, David Firth, Marina Shapira, Michael Stadelmann, and expm-developers@lists.R-forge.R-project.org. *expm: Matrix exponential*, 2014. R package version 0.99-1.1.

[65] John C. Gower. A comparison of some methods of cluster analysis. *Biometrics*, 23(4):623–637, 1967.

[66] W3C HTML Working Group. *HTML5. A vocabulary and associated APIs for HTML and XHTML*, Feb 2014.

[67] Thomas Gruber. Toward principles for the design of ontologies used for knowledge sharing. *International Journal Human-Computer Studies*, 43(5–6):907–928, 1995.

[68] Thomas Gruber. Ontology. In Ling Liu and M. Tamer Özsu, editors, *Encyclopedia of Database Systems*. Springer–Verlag, 2008.

[69] Joseph F. Hair, William C. Black, Barry J. Babin, and Rolph E. Anderson. *Multivariate Data Analysis*. Prentice Hall, 7th edition, 2010.

[70] Paul R. Halmos. *Naive Set Theory*. Undergraduate Texts in Mathematics. Springer, 1960.

[71] Kasper Daniel Hansen, Jeff Gentry, li Long, Robert Gentleman, Seth Falcon, Florian Hahne, and Deepayan Sarkar. *Rgraphviz: Provides plotting capabilities for R graph objects*, 2014. R package version 2.8.1.

[72] Sébastien Harispe, Sylvie Ranwez, Stefan Janaqi, and Jacky Montmain. Semantic measures for the comparison of units of language, concepts or entities from text and knowledge base analysis. *arXiv*, 1(1310.1285), 2013.

[73] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning. Data Mining, Inference, and Prediction.* Springer Series in Statistics. Springer New York, 2009.

[74] John Hebeler, Matthew Fisher, Ryan Blace, Andrew Perez-Lopez, and Mike Dean. *Semantic Web Programming.* Wiley, 2011.

[75] Christian Hennig. *fpc: Flexible procedures for clustering*, 2014. R package version 2.1-7.

[76] Stefan Hinz, Paul DuBois, Jonathan Stephens, Philip Olson, Daniel Price, Daniel So, and Edward Gilmore. *MySQL Reference Manuals*, 2014.

[77] Birger Hjørland and Hanne Albrechtsen. Toward a new horizon in information science: Domain-analysis. *Journal of the American Society for Information Science*, 46(6):400–425, 1995.

[78] Zhi-Liang Hu, Jie Bao, and James M. Reecy. CateGOrizer: A Web-Based Program to Batch Analyze Gene Ontology Classification Categories. *Online Journal of Bioinformatics*, 9(2):108–112, 2008.

[79] Da Wei Huang, Brad T. Sherman, and Richard A. Lempicki. Bioinformatics enrichment tools: paths toward the comprehensive functional analysis of large gene lists. *Nucleic Acids Res.*, 37(1):1–13, 2009.

[80] Da Wei Huang, Brad T. Sherman, and Richard A Lempicki. Systematic and integrative analysis of large gene lists using DAVID bioinformatics resources. *Nature Protocols*, 4(1):44–57, 2009.

[81] Paul Jaccard. Étude comparative de la distribuition florale dans une portion des alpes et des jura. *Bulletin de la Société Vaudoise des Sciences Naturelles*, 37:547–579, 1901.

[82] Jay J. Jiang and David W. Conrath. Semantic similarity based on corpus statistics and lexical taxonomy. In *Proceedings of International Conference Research on Computational Linguistics (ROCLING X)*, pages 19–33, 1997. Tapei, Taiwan.

[83] Cliff A. Joslyn and William J. Bruno. Weighted pseudo-distances for categorization in semantic hierarchies. In Frithjof Dau, Marie-Laure Mugnier, and Gerd Stumme, editors, *Conceptual Structures: Common Semantics for Sharing Knowledge*, volume 3596 of *Lecture Notes in Computer Science*, pages 381–395. Springer Berlin Heidelberg, 2005.

[84] Cliff A. Joslyn. Poset ontologies and concept lattices as semantic hierarchies. In *Lecture Notes in Artificial Intelligence*. Springer-Verlag, 2004.

[85] Cliff A. Joslyn, Susan M. Mniszewski, Andy W. Fulmer, and Gary G. Heaton. The gene ontology categorizer. *Bioinformatics*, 20(s1):169–77, 2004.

[86] Minoru Kanehisa, Susumu Goto, Yoko Sato, Masayuki Kawashima, Miho Furumichi, and Mao Tanabe. Data, information, knowledge and principle: back to metabolism in KEGG. *Nucleic Acids Res.*, 42(Database issue):199–205, 2014.

[87] Leonard Kaufman and Peter J. Rousseeuw. *Finding Groups in Data: An Introduction to Cluster Analysis*. Wiley Series in Probability and Statistics. Wiley, 2005.

[88] Purvesh Khatri, Pratik Bhavsar, Gagandeep Bawa, and Sorin Draghici. Onto-Tools: an ensemble of web-accessible, ontology-based tools for the functional design and interpretation of high-throughput gene expression experiments. *Nucleic Acids Research*, 32(suppl 2):W449–W456, 2004.

[89] Purvesh Khatri and Sorin Drăghici. Ontological analysis of gene expression data: current tools, limitations, and open problems. *Bioinformatics*, 21(18):3587–3595, 2005.

[90] Eric D. Kolaczyk. *Statistical Analysis of Network Data. Methods and Models*. Springer Series in Statistics. Springer New York, 2009.

[91] Joseph B. Kruskal. Multidimensional Scaling by Optimizing Goodness of Fit To a Nonmetric Hypothesis. *Psychometrika*, 29(1):1–27, 1964.

[92] Pierre Legendre. Comparison of permutation methods for the partial correlation and partial mantel tests. *Journal of Statistical Computation and Simulation*, 67:37–73, 2000.

[93] Pierre Legendre and Louis Legendre. *Numerical Ecology*. Developments in Environmental Modelling. Elsevier Science, 1998.

[94] Jim Lemon. Plotrix: a package in the red light district of r. *R-News*, 6(4):8–12, 2006.

[95] Cornellius T. Leondes. *Knowledge-Based Systems, Four-Volume Set: Techniques and Applications*. Elsevier Science, 2000.

[96] John Lim. *ADOdb Library for PHP Manual*, v5.18 edition, Sep 2012.

[97] Dekang Lin. An information-theoretic definition of similarity. In *Proceedings of the Fifteenth International Conference on Machine Learning*, pages 296–304. Morgan Kaufmann Publishers, 1998.

[98] P.W. Lord, R.D. Stevens, A. Brass, and C.A. Goble. Investigating semantic similarity measures across the gene ontology: the relationship between sequences and annotation. *Bioinformatics*, 19(10):1275–1283, 2003.

[99] Steven Maere, Karel Heymans, and Martin Kuiper. BiNGO: a Cytoscape plugin to assess overrepresentation of Gene Ontology categories in Biological Networks. *Bioinformatics*, 21(16):3448–3449, 2005.

[100] Bryan F.J. Manly. *Randomization, Bootstrap and Monte Carlo Methods in Biology.* Chapman & Hall Texts in Statistical Science Series. Taylor & Francis, third edition, 2006.

[101] Nathan Mantel. The detection of disease clustering and a generalized regression approach. *Cancer Research*, 27(2 Part 1):209–220, 1967.

[102] Nathan Mantel and Ranchhodbhai S. Valand. A technique of nonparametric multivariate analysis. *Biometrics*, 26(3):547–558, 1970.

[103] Kantilal V. Mardia, John T. Kent, and John M. Bibby. *Multivariate Analysis.* Probability and Mathematical Statistics. Academic Press, 1979.

[104] David Martin, Christine Brun, Elisabeth Remy, Pierre Mouren, Denis Thieffry, and Bernard Jacq. GOToolBox: functional analysis of gene datasets based on Gene Ontology. *Genome Biology*, 5(R101), 2004.

[105] Luis A. Martínez-Cruz, Angel Rubio, María L. Martínez-Chantar, Alberto Labarga, Isabel Barrio, Adam Podhorski, Víctor Segura, José L. Sevilla Campo, Matías A. Avila, and Jose M. Mato. GARBAN: genomic analysis and rapid biological annotation of cDNA microarray and proteomic data. *Bioinformatics*, 19(16):2158–2160, 2003.

[106] Marco Masseroli, Dario Martucci, and Francesco Pinciroli. GFINDer: Genome Function INtegrated discoverer through dynamic annotation, statistical analysis, and mining. *Nucleic Acids Research*, 32(suppl 2):W293–W300, 2004.

[107] Gaston K. Mazandu and Nicola J. Mulder. Information content-based gene ontology semantic similarity approaches: Toward a unified framework theory. *BioMed Research International*, (292063):1–11, 2013.

[108] David Sawyer McFarland. *JavaScript: The Missing Manual.* O'Reilly Media – Pogue Press, 2008.

[109] Elizabeth M. Miller, Sergio Freire, and Aaron R. Wheeler. Proteomics in microfluidic devices. In Dongqing Li, editor, *Encyclopedia of Microfluidics and Nanofluidics*, volume 3, pages 1749–1758. Springer US, 2008.

[110] Saif M. Mohammad and Graeme Hirst. Distributional measures as proxies for semantic relatedness. *CoRR*, abs/1203.1, 2012.

[111] Jose Luis Mosquera and Alex Sànchez-Pla. SerbGO: searching for the best GO tool. *Nucleic Acids Res.*, 36(Web Server issue):W368–371, 2008.

[112] David Mount. *Bioinformatics: Sequence and Genome Analysis.* Cold Spring Harbor Laboratory Press, 2nd edition, 2004.

[113] M. Mraz, K. Malinova, J. Kotaskova, S. Pavlova, B. Tichy, J. Malcikova, K. Stano Kozubik, J. Smardova, Y. Brychtova, M. Doubek, M. Trbusek, J. Mayer, and S. Pospisilova. miR-34a, miR-29c and miR-17-5p are downregulated in CLL patients with TP53 abnormalities. *Leukemia*, 23(6):1159–1163, 2009.

[114] Darren A. Natale, Cecilia N. Arighi, Winona C. Barker, Judith A. Blake, Carol J. Bult, Michael Caudy, Harold J. Drabkin, Peter D'Eustachio, Alexei V. Evsikov, Hongzhan Huang, Jules Nchoutmboube, Natalia V. Roberts, Barry Smith, Jian Zhang, and Cathy H. Wu. The protein ontology: a structured representation of protein forms and complexes. *Nucleic Acids Research*, 39(Database issue):D539–D545, 2011.

[115] Gottfried E. Noether. *Introduction to Statistics. The Nonparametric Way.* Springer-Verlag, 1991.

[116] Norwegian University of Science and Technology. *eGOn v1.0*, 2004.

[117] Natalya F. Noy, Michael Sintek, Stefan Decker, Monica Crubézy, Ray W. Fergerson, and Mark A. Musen. Creating semantic web contents with protégé–2000. In *The Semantic Web*, pages 60–71. IEEE Intelligent Systems, 2001.

[118] Jari Oksanen, F. Guillaume Blanchet, Roeland Kindt, Pierre Legendre, Peter R. Minchin, R. B. O'Hara, Gavin L. Simpson, Peter Solymos, M. Henry H. Stevens, and Helene Wagner. *vegan: Community Ecology Package*, 2013. R package version 2.0-10.

[119] Herve Pages, Marc Carlson, Seth Falcon, and Nianhua Li. *AnnotationDbi: Annotation Database Interface*, 2014. R package version 1.26.0.

[120] C. Pasquier, F. Girardot, K. Jevardat de Fombelle, and R. Christen. THEA: ontology-driven analysis of microarray data. *Bioinformatics*, 20(16):2636–2643, 2004.

[121] Silvio Peroni, David Shotton, and Fabio Vitali. The Live OWL Documentation Environment: A Tool for the Automatic Generation of Ontology Documentation. In *Knowledge Engineering and Knowledge Management*, volume 7603 of *Lecture Notes in Computer Science*, pages 398–412. Springer Berlin Heidelberg, 2012.

[122] Catia Pesquita, Daniel Faria, Andre O. Falcão, Phillip Lord, and Francisco M. Couto. Semantic similarity in biomedical ontologies. *PLoS Computational Biology*, 5(7), 2009.

[123] Giuseppe Pirró and Jérôme Euzenat. A feature and information theoretic framework for semantic similarity and relatedness. In Peter F. Patel-Schneider, Yue Pan, Pascal Hitzler, Peter Mika, Lei Zhang, Jeff Z. Pan, Ian Horrocks, and Birte Glimm, editors, *The Semantic Web - ISWC 2010*, volume 6496 of *Lecture Notes in Computer Science*, pages 615–630. Springer Berlin Heidelberg, 2010.

[124] Giuseppe Pirró and Nuno Seco. Design, implementation and evaluation of a new semantic similarity metric combining features and intrinsic information content. In Robert Meersman and Zahir Tari, editors, *On the Move to Meaningful Internet Systems: OTM 2008*, volume 5332 of *Lecture Notes in Computer Science*, pages 1271–1288. Springer Berlin Heidelberg, 2008.

[125] David Poole. *Linear Algebra: A Modern Introduction*. Available Titles CengageNOW Series. Thomson Brooks/Cole, 2006.

[126] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2013.

[127] Roy Rada, Hafedh Mili, Ellen Bicknell, and Maria Blettner. Development and application of a metric on semantic nets. *Systems, Man and Cybernetics, IEEE Transactions on*, 19(1):17–30, 1989.

[128] Philip Resnik. Using information content to evaluate semantic similarity in a taxonomy. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, volume 1 of *IJCAI*, pages 448–453. Morgan Kaufmann Publishers Inc., 1995.

[129] Peter N. Robinson, Andreas Wollstein, Ulrike Böhme, and Brad Beattie. Ontologizing gene-expression microarray data: characterizing clusters with Gene Ontology. *Bioinformatics*, 20(6):979–981, 2004.

[130] Fred Ruissen and Frank Baas. Serial analysis of gene expression (sage). In Paul B. Fisher, editor, *Cancer Genomics and Proteomics*, volume 383 of *Methods in Molecularbiology*, pages 41–66. Humana Press, 2007.

[131] Alex Sánchez and Jose Luis Mosquera. The quest for biological significance. In Luis L. Bonilla, Miguel Moscoso, Gloria Platero, and Jose M. Vega, editors, *Progress in Industrial Mathematics at ECMI 2006*, volume 12 of *Mathematics in Industry*, pages 566–570. Springer Berlin Heidelberg, 2008.

[132] Mark Schena. *DNA microarrays: A practical Approach.* The practical Approach Series. Oxford University Press, 1999.

[133] Mark Schena. *Microarray Biochip Technology.* Eaton, 2000.

[134] Andreas Schlicker, Francisco S. Domingues, Jörg Rahnenführer, and Thomas Lengauer. A new measure for functional similarity of gene products based on Gene Ontology. *BMC Bioinformatics*, 7:302, 2006.

[135] Bernd S.W. Schröder. *Ordered Sets. An Introduction.* Birkhäuser, 2003.

[136] Paola Sebastiani, Emanuela Gussoni, Isaac S. Kohane, and Marco Ramoni. Statistical challenges in functional genomics. *Statistical Science*, 18(1):33–60, 2003.

[137] N.H. Shah and N.V. Fedoroff. CLENCH: a program for calculating Cluster ENrichment using the Gene Ontology. *Bioinformatics*, 20(7):1196–1197, 2004.

[138] Roger N. Shepard. The analysis of proximities: Multidimensional scaling with an unknown distance function. i. *Psychometrika*, 27(2):125–140, 1962.

[139] Roger N. Shepard. The analysis of proximities: Multidimensional scaling with an unknown distance function. ii. *Psychometrika*, 27(3):219–246, 1962.

[140] Jeremy G. Siek, Lie-Quan Lee, and Andrew Lumsdaine. *The Boost Graph Library: User Guide and Reference Manual.* Addison Wesley Professional, 2002.

[141] Richard M. Simon, Edward L. Korn, Lisa M. McShane, Michael D. Radmacher, George W. Wright, and Yingdong Zhao. *Design and Analysis of DNA Microarray Investigations.* Springer-Verlag, New York, 2004.

[142] Peter E. Smouse, Jeffrey C. Long, and Robert R. Sokal. Multiple regression and correlation extensions of the mantel test of matrix correspondence. *Systematic Zoology*, 35(4):627–632, 1986.

[143] Robert R. Sokal and Charles D. Michener. *A Statistical Method for Evaluating Systematic Relationships.* University of Kansas science bulletin. University of Kansas, 1958.

[144] Robert Stevens, Chris Wroe, Phillip W. Lord, and Carole A. Goble. Ontologies in bioinformatics. In Steffen Staab and Rudi Studer, editors, *Handbook on Ontologies*, International Handbooks on Information Systems, pages 635–658. Springer, 2004.

[145] G. W. Stewart. *Matrix Algorithms: Basic Decompositions*, volume 1. Society for Industrial and Applied Mathematics, 1998.

[146] Aravind Subramanian, Pablo Tamayo, Vamsi K. Mootha, Sayan Mukherjee, Benjamin L. Ebert, Michael A. Gillette, Amanda Paulovich, Scott L. Pomeroy, Todd R. Golub, Eric S. Lander, and Jill P. Mesirov. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of National Academy of Sciences U.S.A.*, 102(43):15545–15550, 2005.

[147] Pang-Ning Tan, Michael Steinbach, and Vipin Kumar. *Introduction to Data Mining.* Addison-Wesley Longman Publishing Co., Inc., firt edition edition, 2005.

[148] The Gene Ontology Consortium. Gene ontology: Tool for the unification of biology. *Nature Genetics*, 25(1):25–29, 2000.

[149] The Gene Ontology Consortium. Creating the gene ontology resource: Design and implementation. *Genome Research*, 11(8):1425–1433, 2001.

[150] The Gene Ontology Consortium. The gene ontology (go) database and informatics resource. *Nucleic Acids Research*, 32(Suppl. 1):D258–D261, 2004.

[151] The GO Consortium. Gene Ontology Documentation. http://www.geneontology.org/GO.contents.doc.shtml.

[152] The GO Consortium. The GO consortium list. http://www.geneontology.org/GO.consortiumlist.shtml.

[153] Warren S. Torgerson. Multidimensional scaling: I. theory and method. *Psychometrika*, 17(4):401–419, 1952.

[154] Warren S. Torgerson. *Theory and Methods of Scaling*. Wiley, 1958.

[155] William Thomas Tutte. *Graph Theory*. Cambridge Mathematical Library. Cambridge University Press, 2001.

[156] William N. Venables and Brian D. Ripley. *Modern Applied Statistics with S*. Springer, New York, fourth edition, 2002.

[157] James Z. Wang1, Zhidian Du, Rapeeporn Payattakool, Philip S. Yu, and Chin-Fu Chen. A new method to measure the semantic similarity of go terms. *Bioinformatics*, 23(10):1274–1281, 2007.

[158] Open Biomedical Ontologies website. The open biological and biomedical ontologies. http://www.obofoundry.org/.

[159] M. M. Weiss, M. A. Hermsen, G. A. Meijer, N. C. van Grieken, J. P. Baak, E. J. Kuipers, and P. J. van Diest. Comparative genomic hybridisation. *Molecular Pathology*, 52(5):243–251, 1999.

[160] Florian Wickelmaier. An introduction to mds. Technical Report 7, Sound Quality Research Unit, Aalborg University, 2003.

[161] Hadley Wickham. The Split-Apply-Combine Strategy for Data Analysis. *Journal of Statistical Software*, 40(1):1–29, 2011.

[162] Robin J. Wilson. *Introduction to graph theory*. Longman, 1996.

[163] Zeshui Xu. A method based on distance measure for interval-valued intuitionistic fuzzy group decision making. *Information Sciences*, 180(1):181–190, 2010.

[164] A. Young, N. Whitehouse, J. Cho, and C. Shaw. OntologyTraveser: an R package for GO analysis. *Bioinformatics*, 21(2):275–276, 2005.

[165] Guangchuang Yu, Fei Li, Yide Qin, Xiaochen Bo, Yibo Wu, and Shengqi Wang. GOSemSim: an R package for measuring semantic similarity among GO terms and gene products. *Bioinformatics*, 26(7):976–978, 2010.

[166] Barry Zeeberg, Weimin Feng, Geoffrey Wang, May Wang, Anthony Fojo, Margot Sunshine, Sudarshan Narasimhan, David Kane, William Reinhold, Samir Lababidi, Kimberly Bussey, Joseph Riss, J Barrett, and John Weinstein. GoMiner: a resource for biological interpretation of genomic and proteomic data. *Genome Biology*, 4(4):R28, 2003.

[167] Günther Zehetner. OntoBlast function: from sequence similarities directly to potential functional annotations by ontology terms. *Nucleic Acids Research*, 31(13):3799–3803, 2003.

[168] Bing Zhang, Denise Schmoyer, Stefan Kirov, and Jay Snoddy. GOTree Machine (GOTM):a web-based platform for interpreting sets of interesting genes using Gene Ontology hierarchies. *BMC Bioinformativcs*, 5(16), 2004.

[169] Sheng Zhong, Lu Tian, Cheng Li, Kai-Florian Storch, and Wing H. Wong. Comparative analysis of gene sets in the gene ontology space under the multiple hypothesis testing framework. In *Proceedings of the 2004 IEEE Computational Systems Bioinformatics Conference (CSB 2004)*, 2004. IEEE Computer Society.

Index

Part V

Resum en Català

8.1 Introducció

L'arribada dels estudis omics ha generat una quantitat ingent d'informació que ha fet avançar enormement la biologia molecular. Això ha estat possible gràcies a una nova generació de tecnologies anomenades *tecnologies d'alt rendiment* (p.e. microarrays, Next Generation Sequencing (NGS), espectrometria de masses, etc.). Aquestes tecnologies permeten analitzar simultàniament, de forma rutinària, el comportament de milers de característiques (p.e. gens, mRNAs, proteïnes o metabòlits) sota diferents condicions.

Freqüentment, el resultat d'aquests experiments són una llarga llista de característiques que han estat seleccionades en base a algun criteri estadístic (p.e. en un experiment de microarrays, el test-t permet identificar gens diferencialment expressats entre dues condicions experimentals). Amb aquesta llista de gens en mà, l'investigador es troba amb el repte de donar-li una *interpretació biològica*.

Per a donar resposta a aquest repte, usualment, es procedeix en fer un canvi d'enfoc passant de la "significació estadística" a la "significació biològica".

8.1.1 El concepte de la significació biològica

Mentre hi ha un clar consens sobre que vol dir la *Significació Estadística*, no és tant evident que s'ha d'entedre per *Significació Biològica*. Per exemple, alguns autors ([41]) suggereixen que per a entendre la rellevància biològica, donada una llista de característiques (p.e. gens expressats diferencialment), és necessari comparar estadísticament les diferències entre la distribució de les anotacions associades aquestes característiques i l'univers d'anotacions en un espai d'anotacions específic. En canvi, altres autors entenen la Significació Biològica com la forma de caracteritzar la biologia involucrada en un experiment en particular.

8.1.2 La Ontologia Gènica

La tasca d'interpretar biològicament els resultats d'un experiment òmic sol basar-se freqüentment en l'ús de recursos d'anotació existents ([56], [86], [35]). El problema d'això és que freqüentment aquests recursos es centren en un espècie concreta, o depenen d'algun tipus de context específic, o el format de les anotacions no està suficientment orgnanitzat com per a explotar-los de forma sistemàtica i automatitzada. Per tal de solucionar aquests inconvenients, la comunitat científica ha desenvolupat diversos recursos d'anotacions basats en onologies, és a dir, les ontologies biològiques ([158]).

Una *ontologia* és un vocabulari organitzat que cobreix un domini conceptual. Els termes d'aquest vocabulari han d'estar ben definits i allotjats en una estructura jeràrquica de relacions. Al respecte, possiblement la ontologia biològica amb més èxit per a portar a terme interpretacions biològiques és la *Ontologia Gènica* (o *Gene Ontology* (GO)) ([148]).

El projecte de la GO, tutelat pel *The Gene Ontology Consortium* ([152]), intenta anotar els gens i els seus productes gènics amb un conjunt de propietats limitat. És a dir, la GO és un recurs que organitza un vocabulari per anotar gens. Aquest vocabulari cobreix tres grans dominis d'anotacions:

- *Components Cel.lulars* (CC): parts d'una cèl.lula o el seu entorn extracel.lular.
- *Funcions Moleculars* (MF): activitats elementals d'un producte gènic a nivell molecular (p.e. la vinculació o *binding*, la catàlisi, etc.).
- *Processos Biològics* (BP): operacions o conjunts d'esdeveniments moleculars amb un inici i final definits, pertinents al funcionament de les unitats de vida integrada (és a dir, cèl.lules, teixits, òrgans i organismes).

La forma d'entendre aquesta organització és pensar en que els individus (els gens) tenen unes tasques (les funcions) que treballan de manera conjunta per a aconseguir diferents objectius (els processos).

En l'actualitat, a la GO hi han anotats aproximadament 40.000 termes ([22]).

Tal com manen els prinicipis ontològics, els termes de la GO (o termes GO) de cada domini estan disposats jeràrquicament. Així, les relacions conecten les anotacions des del terme menys especialitzat (o arrel) als termes més especialitzats. Hi ha dos grans tipus de relacions:

- **és-un**: estableix una relació entre un terme *pare* i un terme *fill*.
- **part-de**: estableix una relació entre una *part* i l'*entorn*.

No és doncs d'estrenyar que, freqüentment, es descrigui l'estructura de la GO en forma de graf ([122], [72]), on cada terme és representat per un node i cada relació per una aresta.

8.1.3 De la significació biològica a l'estadística

En el camp de la bioinformàtica, de cara a proporcionar interpretacions biolòqiques, s'han desenvolupat diferents estratègies i mètodes [58]. Dues d'aquestes aproximacions basades en la GO són les *mesures de similaritat semàntica* i l'*anàlisi d'enriquiment.*

8.1.3.1 La filosofia de les mesures de la similaritat semàntica

Hem esmentat més amunt 8.1.2 que sovint la GO es defineix com un graf. La *teoria de grafs* ([17], [162], [43], [155]) permet definir diferents tipus de mètriques dirigides a mesurar el "grau" de relació que hi ha entre els nodes del graf. És a dir, aquestes mesures permenten quantificar la *distància* existent entre dos nodes. O dit d'una altra manera, permenten mesurar com de lluny o de prop es troben situats dos termes específics a dintre de l'estructura topològica de la ontologia. No obstant, el concepte de distància és difícil de digerir quan estem parlant d'interpretació biolóqica. Una mesura alternativa més intuitiva que se sol utilitzar és la *mesura de similaritat.* Una similaritat es pot interpretar com a una mena d'inversa de la distància (no en un sentit estricte de funció inversa matemàtica). Per tant, *quant més s'assemblin dos conceptes, la seva similaritat serà més gran i la distància entre ells serà menor.* Ara be, quan un investigador necessita "llegir" què, com i per què es dóna un fenomen, el concepte de similaritat de per si sol no va més enllà. En altres paraules, una mesura de similaritat "simple" pot dir quin és nivell de relació que hi ha entre dos termes, però no és capaç de distingir com de diferents són aquests conceptes en termes *semàntics.*

Les ontologies, en front de les bases de dades relacionals permeten construir "frases" amb un subjecte (els termes), un verb (el tipus de relació) i un predicat (les restriccions sobre les relacions), és a dir, les ontologies permeten crear una lingüistica.

En l'anàlisi lingüístic d'un conjunt de termes estructurats en una d'ontologia, hi ha un gran nombre de mètriques que permeten calcular el nivell de similaritat del contingut sintàctic d'aquests termes basats en l'afinitat de la seva significació ([67], [95], [10]). Entre aquestes mètriques hi han les *mesures de similaritat semàntica.* Per tant, no es d'estranyar que aquest tipus de mètriques hagin estat ben acceptades pels estadístics i els bioinformàtics quan volen estudiar les ontologies biològiques i proporcionar significació biològica.

Hi ha molts mètodes per a mesurar la similaritat semàntica entre dos termes i, de fet, s'han proposat diferents tipus de classificacions d'aquests mètodes ([72]). No obstant, la forma de classificació més acceptada es basa en els elements que composen el graf de la ontologia ([122]). Aquesta classificació consta de tres grans tipus d'estratègies que són:

- l'*aproximació basada en nodes*: examina com de similars són dos conceptes tenint en compte les propietats que s'atribueixen als propis termes, als ancestres i/o els descendents.
- L'*aproximació basada en arestes*: calcula la similaritat en base al nombre de d'arestes (relacions) que hi ha entre els dos nodes comparats.
- l'*aproximació basada en híbrids*: mesura la similaritat combinant mesures basades en nodes i mesures basades en arestes.

8.1.3.2 La filosofia de l'anàlisi d'enriquiment

Un enfoc completament diferent a les mesures de la similaritat semàntica són els *mètodes d'anàlisi d'enriquiment* ([79]). Aquests mètodes permeten avaluar estadísticament si un anotació o un conjunt d'anotacions associades a un o més gens són significativament rellevants. És a dir, el principi bàsic de l'anàlisi és veure si una o més funcions, processos o components no són "normals" de per si sols en un experiment. Concretament, donat un conjunt de gens que cooperen conjuntament és d'esperar que aquestes anotacions mostrin una probabilitat més alta de ser seleccionades. En aquest cas, potencialment, aquestes característiques haurien de ser més rellevants o haurien d'estar enriquides. Per tant, en comptes de cercar que vol dir biològicament un gen específic, la idea és trobar com d'enriquit està un grup de gens en base a la versemblança de l'anotació associada al fenomen d'interès.

De manera general, per portar a terme aquesta tasca, la majoria dels mètodes i eines ([47], [79], [9]) treballen en dues etapes sistemàtiques. Donada la llista de gens seleccionats a dintre d'una població (coneguda com a univers): primer, assignen a cada gen seleccionat totes les anotacions associades a ell i, després, quantifiquen l'enriquiment dels gens anotats en cada categoria (o terme), comparant la proporció de gens d'interès que han estat assignats a aquesta categoria amb la proporció de gens de l'univers que han estat assignats a la mateixa categoria (figura 1.8).

Durant la darrera dècada s'han desenvolupat molts mètodes i eines per a l'anàlisi d'enriquiment, i s'han fet esforços de classificació en base a la seva aproximació metodològica ([47], [79]). Així, s'han proposat tres grans grups d'estratégies:

- L'*Anàlisi d'Enriquiment Singular* (SEA) ([89], [45]): consisteix en estudiar l'enriquiment terme a terme i de manera independentment. Aquesta és l'aproximació més utilitzada.
- L'*Anàlisi d'Enriquiment Modular* (MEA) ([79], [80]): es basa en la idea del SEA, però afegeix mètodes d'anàlisi de xarxes amb l'objectiu de trobar relacions entre grups de gens, per a reorganitzar les co-ocurrences complexes d'anotacions extretes de múltiples espais d'anotació, i mesurar la seva concordància o associació.
- L'*Anàlisi d'Enriquiment de Conjunts de Gens* (GSEA) ([146]): és una idea completament diferent a les anteriors. Aquesta aproximació té en compte la magnitud de les diferències entre les condicions de mesura per a cada gen resultant de l'experiment òmic. La idea consisteix en testar l'enriquiment d'algun conjunt de gens predefinit (recollits en diferents bases de dades i estudis computacionals) respecte dels gens de l'experiment.

8.2 Hipòtesis, objectius i organització de la tesi

8.2.1 Les hipòtesis

Les dues aproximacions presentades a 8.1.3, han estat i són de gran ajuda per a estudiar la *Significació Biològica.* No obstant, ambdues concepcions presenten algunes debilitats i qüestions fonamentals que no han estat encara estudiades. Concretament:

1. Respecte a les similaritats semàntiques:

   (a) No hi han demostracions que provin que aquests tipus de similaritats són realment mesures de similaritat, entenent-les com a revers complementari de les distàncies mètriques.

   (b) Hi ha un dèbil enllaç entre les similaritats semàntiques dels mètodes basats en nodes i els mètodes basats en arestes quan són aplicades al cas de la GO.

2. Respecte als mètodes i les eines GO per a l'anàlisi d'enriquiment:

   (a) Quan un investigador vol trobar un programari que li permeti fer una anàlisi d'enriquiment, és quasi segur que es perdi o no trobi l'eina més adequada pels seus objectius degut a l'alt nombre d'eines GO existents, fins i tot havent estat classificades ([47], [79]).

   (b) La velocitat de desenvolupament de nous mètodes i eines per a l'anàlisi d'enriquiment, així com la millora de les aplicacions existents és considerable. Al respecte, no s'ha portat a terme cap seguiment exhaustiu de les capacitats de les aplicacions, ni tampoc s'ha definit una estratègia general que permeti desenvolupar noves eines que cobreixin mancasses existents.

Amb el repte d'aportar llum a aquestes qüestions, els objectius principals i específics d'aquesta tesi es presenten a continuació.

8.2.2 Els objectius

El context d'aquesta tesi està centrat en els mètodes i les eines que s'utilitzen per a atribuir interpretació biològica a dades generades amb tecnologies d'alt rendiment en experiments òmics, a través del discurs de la *Gene Ontology.*

8.2.2.1 Els objectius principals

En aquesta tesi es plantegen dos objectius principals:

1. L'estudi de dos tipus de mesures de similaritat semàntica per a explorar categories GO.

2. La classificació i estudi de les eines GO per a l'anàlisi d'enriquiment.

8.2.2.2 Els objectius específics

Per tal aconseguir donar resposta als dos objectius principals es presenten a continuació dues llistes d'objectius específics:

1. Objectius específics associats amb l'estudi de les mesures de similaritat semàntica:

(a) Demostrar que ambdues mesures estan relacionades amb el concepte de distància mètrica.

(b) Desenvolupar un paquet de `R` per a calcular similaritats semàntiques entre termes d'una ontologia arbitrària i comparar perfils de similaritat semàntica.

2. Objectius específics associats amb la classificació i estudi de les eines GO per a l'anàlisi d'enriquiment:

(a) Definir una llista de funcionalitats que permeti classificar les eines GO per a l'anàlisi d'enriquiment.

(b) Classificar les eines GO existents en base a la llista de funcionalitats definida i d'acord a les seves capacitats.

(c) Desenvolupar una eina web dirigida a seleccionar i comparar les eines GO que millor s'adaptin a les necessitats de l'usuari.

(d) Estudiar l'evolució de les eines GO per tal de caracteritzar l'existència de models representatius.

(e) Desenvolupar una ontologia per a organitzar un vocabulari dirigit al desenvolupament de noves eines GO.

8.2.3 La organització

La tesi està dividida en quatre parts. Desprès d'aquesta introducció i formulació del problema, la segona part està adreçada a la recerca de l'estudi de les dues mesures de similaritat semàntica. Aquesta part està estructurada en tres seccions. La primera introdueix els materials i mètodes utilitzats. La segona presenta els resultats obtinguts. I la tercera discuteix les contribucions aportades. La tercera part de la tesi està adreçada a la recerca de l'estudi de les eines GO per a l'anàlisi d'enriquiment. Aquesta part està estructurada en tres seccions, de forma similar a la de la segona part. I finalment, a la quarta part es presenten les conclusions de la tesi. (figura 8.1).

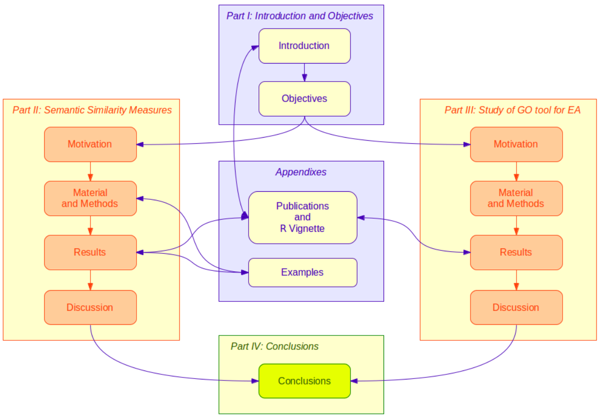


Figure 8.1: Organització de les diferents parts que composen la tesi i les relacions entre elles.

8.3 Estudi de dues mesures de similaritat semàntica per a explorar categories GO

8.3.1 Materials i mètodes

8.3.1.1 Conceptes fonamentals de la Teoria de Grafs

Hi han diferents tipus d'aproximacions que utilitzen la teoria de grafs ([43], [17]) com a base per a definir mètriques i estudiar les relacions existents entre les anotacions, amb l'objectiu final d'aportar intrepretació biològica. Una d'aquestes idees és considerar la GO en forma de graf i usar el concepte de distància (o similaritat) per a comparar els termes GO. Dues d'aquestes aproximacions són d'especial interés pels objectius d'aquesta tesi. La primera és una similaritat semàntica proposada per en Lord *et al.* ([98]), basada en el concepte de *Contingut d'Informació* (IC) ([128]). I la segona és una pseudo-distància proposada per en Joslyn *et al.* ([85]), basada en l'estructura inherent al graf. En ambdós casos és necessari establir els conceptes bàsics sobre la teoria de grafs.

Conceptes bàsics sobre grafs

Un **graf** és una parella de conjunts $G = (V, E)$ formada per **vèrtex** (o **nodes**) V, i **arestes** E que estableixen relacions entre els vèrtex.

És fàcil veure que molts problemes de la vida quotidiana es poden plantejar en forma de graf, essent representats en un pla on un cojunt de punts (els vèrtex) estan units per algunes línies (les arestes), que indican si hi ha una relació explícita entre dos d'aquests punts.

Ara be, de vegades només ens interessa estudiar una "petitat" part de les relacions existents en el graf. En aquests casos, el que es pot fer és considerar només els subconjunt de vértex i d'arestes d'interés per a recuperar el **subgraf** que volem explorar. Tanmateix, en certes situacions només volem estudiar algun tipus de relacions que estan condicionades per alguna forma de restricció. Per exemple, un **graf dirigit** (o **digraph**) ens permet explorar com és un conjunt de nodes i d'arestes als quals se'ls hi ha aplicat una funció que assigna a cada aresta un parell ordenat de vèrtex, és a dir, l'aresta $e_{ij} \in E$ *uneix el* ***node origen*** v_i *amb el* ***node terminal*** v_j.

Un tipus de digrafs especialment important pels nostres objectius, són els DAGs. Un **Graf Dirigit Acíclic** (o DAG) és un digraf que no accepta cap node que es relacioni amb si mateix per una aresta. Els DAGs que tenen un node que és el "pare" de tots els altres nodes i aquest és "orfe", diem que és el node **arrel** (o *root*), i per a indicar aquesta característica diem que es tracta d'un **DAG arrelat** (o *rooted DAG*).

Mesures bàsiques per a descriure un graf

Aquest tipus d'estructures permeten definir camins entre els nodes. Un **camí** és la seqüència natural de vèrtex que hi ha entre el node **origen** i el node **terminal**. Llavors, en el cas de que hi hagi un camí entre dos nodes direm que *el node terminal és assolible des del node inicial.* Notis doncs que donada la noció de camí, sorgeix la primera mesura que permet valorar com de lluny o de prop es troben dos nodes, i és el concepte de **longitud**, és a dir, el nombre d'arestes que hi ha en el camí.

Altres mesures fonamentals per a estudiar com són els nodes i les arestes d'un graf són, per exemple, l'**ordre**, que es defineix com el nombre de vèrtex del graf, o el **grau** d'un vèrtex, que és el nombre d'arestes incidents en ell. Associats al grau, també és comú tenir en compte el **grau d'entrada** d'un node, que és el nombre d'arestes arriben a ell, i el **grau de sortida**, que són el nombre d'arestes que surten d'ell.

Matrius i grafs

Les representacions visuals dels graf són unes eines molt útils per a fer-se una idea molt ràpida de les relacions entre els nodes. Ara be, des del punt de vista analític, de cara a avaluar les relacions entre els nodes del graf, és millor adoptar un punt de vista algebraic. Així doncs, usualment, sol treballar-se amb les formes matricials associades als grafs. Les matrius més rellevants que permeten descriure un graf són:

- La **matriu d'adjacència** d'un graf: els seus elements són el nombre d'arestes entre cada parell de nodes, i quan dos nodes v_i i v_j no estan connectats el valor és 0. La forma de llegir aquesta matriu és fer-ho des de les files (nodes origen) en direcció cap a a les columnes (nodes terminals).

- La **matriu d'incidència** d'un DAG: els elements prenen valor 1, si el node de la fila és node l'origen, valor -1, si el node de la fila és final, o valor 0, altrament.

- La **matriu d'accessibilitat** d'un DAG: els elements prenen valor 1 si el node de la columna és accessible pel node de la fila, o 0 en cas contrari.

8.3.1.2 El graf de la GO

L'estructura de la GO pot ser descrita en termes d'un graf, on els **termes GO** són els nodes del graf, i cada relació entre dos termes GO és una **aresta**. De fet, cadascun dels dominis de la GO (és a dir, CC, MF i BP 8.1.2) és un DAG arrelat. Alguna de la terminologia, usualment, utilitzada és:

- **Pares**: nodes terminals menys especialitzats d'un terme GO específic.

- **Fills**: nodes inicials més especialitzats d'un terme GO menys específic.

- **Ancestres**: tots els nodes menys especialitzats pertanyents als camins existents entre un terme GO específic i el node arrel.

- **Descendència**: tots els nodes més especialitzats pertanyents als camins existents entre un terme GO específic i els termes més específics terminals del DAG.

Com hem vist anteriorment, hi han dos tipus d'arestes (les relacions) entre els termes de la GO. La primera relació s'anomena **és-un**, que estableix una relació entre un *pare* i un *fill*, i la segona s'anomena **part-de**, que estableix una relació entre una *part* i l'*entorn*.

L'estudi de la GO basat en la Teoría de Grafs

L'objectiu de l'*anàlisi de la GO* és facilitar la interpretació biològica a través de les anotacions dels productes gènics. La idea bàsica és que, donada una llista de gens, es cerquen quins són els termes GO específics en el DAG de la GO que els anoten. Cadascun d'aquests gens poden estar anotats en cap, un o més d'un terme GO. Aquest fet determina un *graf induït*. És a dir, un cop seleccionats els termes GO que anoten la llista de gens, es poden recuperar tots els ancestres i les relacions entre ells de manera automàtica. En altres paraules, es pot extreure un subgraf que s'anomena graf induït. Ara be, els grafs induïts poden arribar a ser estructures molt complexes, especialment quan la llista de gens és molt llarga. Per tant, és important disposar d'una bona formalització matemàtica per a manegar correctament aquestes estructures.

8.3.1.3 L'enfoc d'en Carey

En Carey va introduir un formalisme simple ([23]) per a treballar amb ontologies amb propòsits estadístics. La seva idea es basa en el concepte de *refinament* de les relacions, defnint una **relació de refinament 1** entre dos termes t_i i t_j, com una relació tal que no pot exististir un terme t_k que sigui un refinament de t_i i que t_j sigui un refinament de t_k. Per tant, les relacions entre els termes GO s'han d'entendre com a refinaments dels conceptes anotats. Notis doncs que el node arrel està refinat per tots els termes del DAG arrelat de la GO, però aquest no refina a cap altre terme GO.

En base a aquesta filosofia, en Carey redefineix la matriu d'adjacències com la **matriu de refinaments (d'un pas)**, i introdueix la **matriu de mapejat**, que permet relacionar una llista d'objectes amb els termes del DAG de la ontologia que els anoten. Els elements d'aquesta matriu conten un 1 si l'objecte està assignat a un terme específic o 0 en cas contrari.

El principal interès d'en Carey és formalitzar una estructura que permeti relacionar els objectes (els gens) amb el vocabulari (els termes GO) de la ontologia per tal d'extreure la informació o comparar anotacions. Així doncs, defineix un **Object-Ontology Complex** (OOC) com una quaterna ordenada $(\mathcal{T}, \Gamma, \Omega, \mathbf{M})$ on $\mathcal{T}$ és el vocabulari, Γ és una matriu de refinaments codificant la ontologia basada en $\mathcal{T}$, Ω és un conjunt d'identificadors d'objectes, i $\mathbf{M}$ és una matriu de mapejat des de Ω a $\mathcal{T}$. La figura 8.2 mostra una representació gràfica d'un OOC amb 10 objectes i 12 termes.

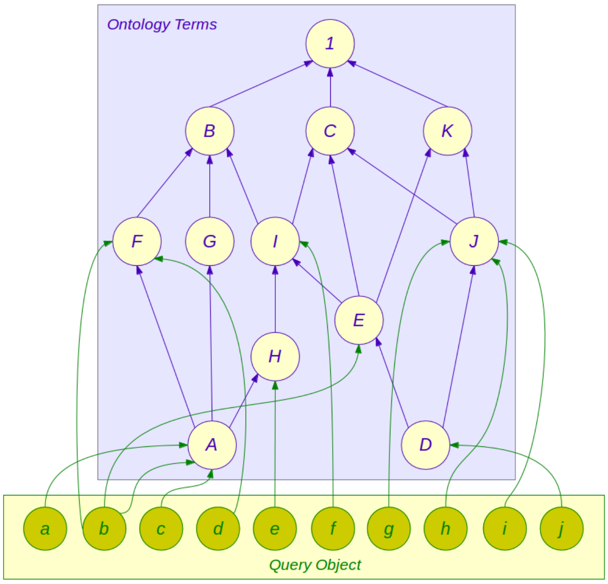


Figure 8.2: Representació d'un OOC amb 10 objectes anotats a una ontologia amb 12 termes.

En base a l'OOC, en Carey introdueix una matriu que permet definir diferents mesures de càlcul, i és la **matriu de cobertura**. Aquesta matriu es pot calcular amb la fórmula

$$\mathbf{C} = \bigoplus_{k=1}^{d_0+1} \mathbf{M}\Gamma^k = (c_{ij})_{p\times n}$$

on c_{ij} és 1 si el terme j-èssim o qualsevol dels seus refinaments està associat amb l'objecte i-èssim (via la matriu de mapejat M), o 0 en cas contrari.

8.3.1.4 Conceptes fonamentals de la Teoria de Conjunts Parcialment Ordenats

Joslyn *et al.* ([85], [84], [83]) van proposar una alternativa a l'enfoc d'en Carey ([23]) basada en els principis de la *Teoria de Conjunts Parcialment Ordenats* ([135], [39], and [51]) de l'algebra.

Un **conjunt parcialment ordenat** (POSET) finit és una estructura matemàtica $\mathcal{P} = \langle P, \leq \rangle$ on P és un conjunt finit i $\leq\subseteq P^2$ és una relació binària anomenada **ordre parcial** on P és tal que la relació és reflexiva, antisimètrica i transitiva es compleixen.

Cada POSET defineix un DAG, i cada DAG determina un POSET basat en l'ordre parcial dels nodes.

Dos nodes $p_i, p_j \in P$ d'un POSET es diu que són **comparables** quan $p_i \sim p_j \Leftrightarrow p_i \leq p_j$ or $p_i \geq p_j$. Basant-se en aquesta noció es pot definir una **cadena** com una col.lecció de nodes comparables. Notis que aquest concepte seria l'equivalent a la noció de camí d'un graf.

Ara, de la mateixa manera que es va presentar les mesures que permeten descriure els nodes i arestes d'un graf, en la teoria dels POSET es poden introduïr coneptes que permeten portar a terme tasques anàlogues. Per exemple, l'**alçada** d'un POSSET es defineix com la grandària de la cadena més llarga, i si C és una cadena finita d'un POSET, la seva **longitud** és $l(C) := |C| - 1$.

Arribat aquest punt en Joslyn *et al.*, introdueixen la noció de la **Ontologia POSET** (POSO). Aquest formalisme es pot considerar com l'equivalent a l'OOC d'en Carey. Una POSO és una estructura $\mathcal{O} = \langle \mathcal{P}, X, F \rangle$ on $\mathcal{P} = \langle P, \leq \rangle$ és un poset, X és un conjunt finit no-buit d'etiquetes, i F és una funció de mapejat.

8.3.1.5 Les mesures de similaritat, distància i similaritat semàntica

La noció més intuitiva per a mesurar com s'assemblen o difereixen dos conceptes és la de similaritat. Una **mesura de similaritat** és una funció que donats dos objectes ω_i i ω_j, els hi assigna un número real s_{ij} entre 0 i 1, de manera que serà 0 si és la similaritat mínima, 1 quan la similaritat és màxima, i $s_{ij} = s_{ji}$. Per tant, es tracta d'una una funció que *quantifica* la similaritat entre dos objectes. Ara be, una mesura de similaritat és pot entendre com una forma de complementari revers de la mesura de distància. Es diu que una mesura és una **distància mètrica** si donats dos objectes ω_i i ω_j, els hi assigna un número real d_{ij} de manera que aquesta funció sigui no negativa, $d_{ij} = 0 \Leftrightarrow \omega_i = \omega_j$, $\forall \omega_i, \omega_j \in \Omega$, simètrica i compleixi la desigualtat triangular ([38]).

Una mesura de similaritat semàntica pot ser considerada com un tipus de mesura de similaritat, però amb certes restriccions, essent la més destacable que les similaritats semàntiques són enteses per a mesurar elements d'una jerarquia ([72]). Ara be, formalment no hi ha una única manera de definir una similaritat semàntica perquè depèn del camp d'estudi, de la taxonomia o del mètode usat ([110], [72], [128], [85],[122]). En qualsevol cas, de manera general es pot considerar que donat un conjunt Ω una **mesura de similaritat semàntica** és una funció que donats dos objectes ω_i i ω_j els hi assigna un número real sim_{ij} de manera que $sim_{ij} \geq a$, $sim_{ij} = b \Leftrightarrow \omega_i = \omega_j$, és una funció simètrica i $sim_{ij} \leq sim_{ii}$.

Hi han diferents tipus de classificacions de similaritat semàntica ([72]), però possiblement la més acceptada és la que les organitza d'acord amb els elements dels graf ([122]).

8.3.1.6 Els terme de la GO i les mesures de similaritat semàntica

En aquesta tesi hem considerat dues aproximacions diferents per a calcular la similaritat entre els termes de la GO. La primera és una mesura de similaritat semàntica proposada per en Lord *et al.* ([98]). Es tracta d'una aproximació basada en nodes. La segona són unes mesures de pseudo-distàncies proposades per en Joslyn *et al.* ([84], [85], [83]) i es tracten d'aproximacions basades en arestes.

La mesura de Lord

La mesura de Lord *et al.* és una mesura basada en el concepte del *Contingut d'Informació* (IC) proposat per en Resnik ([128]). L'IC es defineix com:

$$i(t) = -logP(t)$$

on $P(t)$ és la probabilitat de que un terme del DAG associat a la ontologia sigui seleccionat. Notis doncs que el càlcul de l'IC de cada terme recau en les relacions donades per l'estructura del DAG. Per tant, la informació entre dos termes és usualment proporcional a l'IC de l'*Ancestre Comú Més Informatiu* (MICA) ([128]) en el DAG arrelat. Hi han moltes mesures de similaritat semàntica que es basen en la MICA. Per exemple, en Resnik va proposar que donats dos termes del DAG t_i i t_j, llavors la seva similaritat semàntica es pot calcular com:

$$sim_{Res}(t_i, t_j) = \max_{t \in S(t_i, t_j)} [i(t)]$$

on $S(t_i, t_j)$ és el conjunt del termes que subsumen ambdós termes. Ara be, com el DAG d'una ontologia admet múltiples pares per a cada terme, Lord *et al.* ([98]) van argumentar que aquesta mesura només té en compte un únic ancestre comú i que per tant és millor usar el mínim de la probabilitat de que un terme sigui seleccionat $P(t)$, és a dir,

$$sim_{Lord}(t_i, t_j) = \min_{t \in S(t_i,t_j)} [P(t)].$$

Però en Resnik també havia introduït una segona mesura que considerava la probabilitat en comptes de l'IC:

$$sim_{P(t)}(t_i, t_j) = \max_{t \in S(t_i,t_j)} [1 - P(t)].$$

La mesura de Joslyn

En Joslyn *et al.* ([84], [85], [83]) van proposar les mesures de les pseudo-distàncies. Una **pseudo-distància** és una funció que assigna un número δ_{ij} a una parella de nodes comparables p_i i p_j de manera que aquest valor està afitat per les longituds dels camins més curt i més llarg entre aquests dos nodes. En la seva primera aproximació, Joslyn *et al.* ([85]) van suggerir quatre pseudo-distàncies:

1. La longitud de la cadena mínima
$$\delta_m := h_*.$$
2. La longitud de la cadena màxima
$$\delta_x := h^*.$$
3. La mitjana de les longituds de les cadenes estremes
$$\delta_{ax} := \frac{h_*(p_i,p_j) + h^*(p_i,p_j)}{2}.$$
4. La mitjana de les longituds de totes les cadenes
$$\delta_{ap} := \frac{\sum_{h \in \mathbf{h}(p_i,p_j)} h}{|\mathbf{h}|}.$$

8.3.1.7 `sims`: Un paquet `R` per a calcular similaritats semàntiques d'una ontologia

`sims` és un paquet desenvolupat en `R` ([126]) amb el suport de funcions específiques dels paquets: `AnnotationDbi` ([119]), `expm` ([64]), `GOstats` ([54]), `plyr` ([161]), `Matrix` ([11]), `igraph` ([37]), `methods` ([126]), `plotrix` ([94]), `Rgraphviz` ([71]) i `vegan` ([118]).

8.3.2 Resultats

Els resultats d'aquesta part de la tesi s'han dividit en dos tipus de contribucions: menors i majors. El motiu d'aquesta estructuració són dos fets. El primer és que, durant el procés de desenvolupament del paquet `sims` vam observar que algunes de les parts de les funcions implementades podien ser utilitzades com a demostracions alternatives d'algunes propietats de la teoria de grafs. El segon és que, de vegades a la literatura bioinformàtica s'observa una relaxació en la formalització dels conceptes i això pot induir a males interpretacions o errors, que alhora poden ser passats per alt. Per tant, per tal de suavitzar aquest segon fet i amb la idea d'apuntar alguna idea alternativa hem cregut apropiat presentar aquestes formalitzacions com a contribucions menors.

8.3.2.1 Contribucions menors

A la secció 3.1.3 es va definir un graf simètric com a un graf tal que les arestes connectaven els seus nodes en ambdues direccions. Per tant, gairebé de forma evident, és fàcil veure que la matriu d'accessibilitat associada al graf és una matriu simètrica. Per tant, en totes les funcions implementades en el paquet `sims` que maneguen matrius simètriques, només es va considerar la matriu triangular inferior ([125]) i es va reorganitzar com a vector, reduint així el nombre de la quantitat d'informació que s'ha de manegar computacionalment.

Una de les qüestions més importants en el procés de càlcul de les mesures per a mesurar la similaritat entre els termes és conèixer el nombre de nodes i arestes del DAG. Per exemple, en el càlcul dels IC és necessari conèixer el nombre de vegades que un terme ha estat assolit des d'un objecte que el referencia. Per tant, saber el nombre d'arestes incidents en ell és condició *sine qua non.* En aquest sentit, hi han diferents mètodes per portar a terme aquests càlculs. Dues d'elles són el *Teorema de Handshaking* i el seu corol.lari ([155], [155], [17]) de la teoria de grafs ([43]). Ara bé, durant el proces de desenvolupament del paquet `sims` vam optar per implementar formes de càlcul basades en matrius. Això ens va fer notar que algunes de les funcions implementades suggerien una forma alternativa per a demostrar el citat teorema i el seu corol.lari a partir de la matriu d'incidència del DAG. Fet no observat en la literatura consultada ([155], [155], [17]). En aquest sentit es presenta com a resultat les proves corresponents (veure versió estesa de la tesi 4.1 i 4.1).

Respecte a les mesures de similaritat semàntica, s'ha observat que, com hem comentat a 8.3.2, de vegades hi ha una relaxació en el formalisme a l'hora d'introduir conceptes nous, fet que pot induir a interpretacions inapropiades o mal enteses. Per tant, com veurem en les contribucions majors 8.3.2.2 hem provat dos resultats associats al càlcul dels IC. La primera qüestió que hem matisat és reintrepretar i demostrar el teorema de la propietat monòtona de la teoria de probabilitats ([55]), en termes de l'enfoc d'en Carey. És a dir, donats dos termes de la GO t_i i t_j, si t_i *és-un* (o *part-de*), llavors $P(t_i) \leq P(t_j)$. Aquest fet dóna lloc a enunciar i demostrar que donat que el node arrel t_0 d'una ontologia és el node menys refinat, llavors el IC associat és nul, ja que per la propietat monòtona $P(t_0) = 1$.

8.3.2.2 Contribucions majors

La primera idea que es vol destacar com a una contribució major és que, tot i que els formalismes d'en Carey 8.3.1.3 i en Joslyn *et al.* 8.3.1.4 estan fonamentats en dues teories completament diferents (és a dir, la Teoria de Grafs ([43]) i la Teoria de POSET ([39], [135])), l'OOC i la POSO presenten un cert nivell d'analogia en les seves concepcions respectives. Notis al respecte, que la *Ontologia* de l'OOC és el concepte equivalent al *POSET*, l'*Objecte* és el *conjunt d'objectes* de la POSO i la *matriu de mapejat* de l'OOC és l'anàleg a la *funció de mapejat* entre el conjunt d'objectes i els posets. Per tant, donada una llista de gens generada en un experiment òmic, l'OOC i la POSO són dues estructures que permeten *atribuir significat biològic.*

Una qüestió important per a conèixer la similaritat semàntica entre dos termes és la quantitat d'informació que tenen en comú que comparteixen el termes, que ve donada pels termes que els subsumen. En aquest sentit, en Lord *et al.* ([98]) argumenten que la mesura de similaritat proposada per en Resnik i basada en el MICA ([128]), només té en compte un únic ancestre comú i proposen com a alternativa mesura 8.3.1.6. Ara bé, donat que la funció del logaritme és una funció monòtona creixent, es té que

$$\max_{t \in S(t_i,t_j)} [-logP(t)] = -log \min_{t \in S(t_i,t_j)} [P(t)]$$

i en conseqüència,

$$sim_{Res}(t_i, t_j) = sim_{Lord}(t_i, t_j).$$

S'ha observat que per a calcular els IC dels termes de la ontologia es pot utilitzar el producte matricial de la matriu del nombre de camins de qualsevol longitud entre les parelles dels termes per la matriu de mapejat i així obtindre el nombre de vegades que cada terme o qualsevol de les seves especialitzacions apareix a la ontologia. Per tant, presentem la següent proposició (la demostració es pot veure a la versió estesa de la tesi 4.4):

Proposició: Sigui $(\mathcal{T}, \Gamma, \Omega, \mathbf{M})$ un OOC, on $\mathcal{T}$ és un conjunt de s termes d'un vocabulari, Γ és la matriu de refinaments codificant la ontologia basada en $\mathcal{T}$, Ω és el conjunt de p identificadors d'objectes i $\mathcal{M}$ és la matriu de mapejat de Ω a $\mathcal{T}$. Aleshores la matriu del nombre de vegades que cada terme t_i o qualsevol de les seves especialitzacions referencia a un ancestre específic t_j es pot calcular com

$$\mathbf{N}_t = \mathbf{M}(\mathbf{I} - \Gamma)^{-1}$$

Corol.lari: Sigui $(\mathcal{T}, \Gamma, \Omega, \mathbf{M})$ un OOC. Aleshores, el nombre de vegades que un terme t_j o qualsevol dels seus refinaments apareix en l'OOC es pot calcular sumant les columnes de la matriu N_t,

$$n_j = \sum_{i=1}^{s} n_{ij}$$

on s és el nombre de termes en $\mathcal{T}$.

En Cuadras ([38]) va suggerir que una distància mètrica es pot calcular en termes d'una similaritat amb via $d_{ij} = 1 - sim_{ij}$. Per tant, basant-nos en aquest fet proposem els següents resultat (les demostracions es poden consultar a la versió estesa de la tesi 4.5 i 4.6, respectivament):

Proposició: Donats dos termes t_i i t_j de $\mathcal{T}$ tals que t_i *és-un* t_j aleshores,

$$d(t_i, t_j) = 1 - sim_{P(t)}(t_i, t_j)$$

és una distància mètrica.

Proposició: Siguin $p_i, p_j \in P$ dos nodes comparables tals que $p_i \leq p_j$. Aleshores, la pseudo-distància de la longitud de la cadena mínima és una distància mètrica.

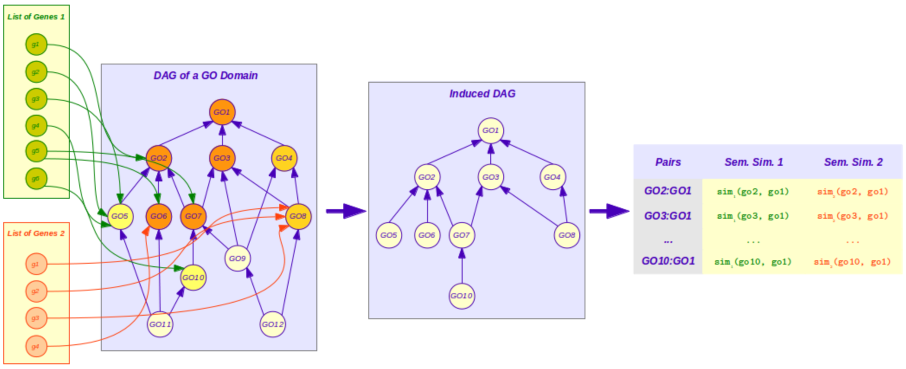
El paquet *sims* és un paquet per a calcular similaritats semàntiques entre els termes d'una ontologia arbitrària. Té implementades 14 tipus de mesures diferents de les aproximacions basades en nodes i de les basades en arestes. La taula 4.2 mostra les mesures implementades.

sims està disponible lliurament, sota una llicència GPL-2 (`http://www.r-project.org/Licenses/GPL-2`), i es pot descarregar des del repositori de `GitHub` `https://github.com/jlmosquera/sims`.

El paquet proporciona un total de 51 funcions, que es poden consultar a la taula 4.1, on estan organitzades segons el tipus de possibilitats que ofereixen. Breument:

1. Funcions per a manegar una ontologia (és a dir un OOC).
2. Funcions per a calcular similaritats semàntiques entre els termes de la ontologia
3. Funcions per a calcular perfils de similaritat semàntica de la GO associats a una llista de Entrez Ids.
4. Funcions per a comparar perfils de similaritat semàntica de la GO associats a una llista de Entrez Ids.

De vegades un investigador necessita comparar dues llistes de gens. Una opció és fer aquesta comparació a través de les anotacions dels gens. En aquest sentit, en el **sims** s'han implementat unes funcions específiques per a portar a terme aquesta tasca en base al que hem anomenat *perfils de similaritat semàtica*. La idea bàsica és que donades dues llistes de `Entrez Gene Ids` es cerquen al domini de la GO corresponent tots els termes anotats per ambdues llistes, es reconstrueix el graf induït i es calculen totes les similaritats semàntiques del graf induït, tant per a una llista com per a l'altra, en base a alguna mesura de similaritat semàntica seleccionada (veure figura 8.3).



Pairs	Sem. Sim. 1	Sem. Sim. 2
GO2:GO1	$\mathtt{sim}_1\mathtt{(go2, go1)}$	$\mathtt{sim}_2\mathtt{(go2, go1)}$
GO3:GO1	$\mathtt{sim}_1\mathtt{(go3, go1)}$	$\mathtt{sim}_2\mathtt{(go3, go1)}$
...	...	...
GO10:GO1	$\mathtt{sim}_1\mathtt{(go10, go1)}$	$\mathtt{sim}_2\mathtt{(go10, go1)}$

Figure 8.3: Esquema per a comparar dos perfils de similaritat semàntica associats amb dues llistes de gens respectivament.

Aleshores, donats els perfils, el `sims` té implementades algunes funcions que permeten comparar-los a través de:

1. Una descriptiva per a cada perfil de similaritat semàntica.
2. Un test de Mantel per a avaluar l'associació entre les matrius de distàncies associades als perfils.
3. La similaritat del cosinus ([147]) per a determinar la similaritat entre els dos perfils.
4. Un histograma que mostra les dues distribucions dels perfils sobre la mateixa figura.
5. Un diagrama de barres, on les barres estan associades a cada parella de termes.
6. El graf induït del domini de la GO associat a una (o les dues) llista(es) de gens.

A l'apèndix A es proporciona una vinyeta del paquet a on es poden trobar exemples de les principals possibilitats del `sims`.

8.3.3 Discussió

La primera part de la tesi estava centrada en dues mesures de similaritat semàntica per a avaluar termes de la *Gene Ontology* (GO) ([148], [149], [150]), amb l'objectiu de donar una interpretació biològica als resultats generats en experiments omics amb dades d'alt rendiment. La recerca intentava mostrar que ambdues aproximacions estan relacionades amb el concepte de distància mètrica, així com desenvolupar un paquet R per calcular mesures de similaritat semàntica entre termes ontològics i proporcionar una manera de comparar perfils de similaritat semàntica.

Hem vist que hi han diverses metodologies per atribuir significat biològic a una llista de gens ([72]) i que la forma en que es sintetitza el mapejat dels gens a la GO és una peça important per a la formalització de les aproximacions ([23], [84]).

La primera aproximació estudiada ha estat una mesura de similaritat semàntica proposada per en Lord *et al.* ([98]). Es tracta d'una aproximació basada en l'estudi del nodes i fonamentada en la Teoria de grafs ([43]). En canvi, la segona aproximació, emmarcada en la Teoria de Conjunts

Parcialment Ordenats (POSET) ([135], [39], and [51]), és una mesura de pseudo-distància proposada per Joslyn *et al.* ([84], [85], [83]), i és una aproximació basada en l'estudi de les arestes.

Per a centrar-nos en l'estudi de la mesura de similaritat semàntica proposada per Lord *et al.*, vam recórrer a una estructura anomenada Object-Ontology Complex (OOC), proposada per en Carey ([23]). L'OOC permet relacionar una llista d'objectes (els gens) amb el vocabulari d'una ontologia (els termes GO) d'una ontologia a través d'una matriu de mapejat que assigna a cada objecte els termes organitzats en un Graf Dirigit Acíclic (DAG) ([43]) associat amb la ontologia (el DAG d'un domini de la GO).

La literatura ofereix una llarga llista de mesures de similaritat semàntica per a estudiar les relacions entre el termes d'una ontologia ([72], [122],[85], [127]). Moltes d'elles recauen en la longitud del camí més curt entre dos termes ([127]). En canvi, la mesura de similaritat semàntica d'en Lord *et al.* es basa en el concepte de Contigut de la Informació (IC) proposat per Resnik ([128]). L'IC és una mesura basada en la probabilitat d'aparició d'un terme de la ontologia. Vam detectar que algunes de les propietats associades a la mesura de l'IC mostraven una pèrdua de claredat. En aquest sentit vam demostrar la propietat monòtona de la probabilitat ([55]) en termes de la concepció d'en Carey i que el node arrel d'una ontologia és el terme amb l'IC mes baix, i que de fet és nul.

D'altra banda, en Resnik va proposar una mesura que depèn de l'Ancestre Comú Més Informatiu (MICA) ([128]), però Lord *et al.* van argumentar que aquesta mesura mesura només té en compte un únic ancestre comú, i van suggerir com a alternativa la seva mesura que depen de la probabilitat mínima d'un terme quan hi ha més d'un pare compartit. Ara be, nosaltres vam demostrar que de fet ambdues mesures són la mateixa mesura de similaritat semàntica.

Durant el procés de desenvolupament del paquet `sims` vam observar que podíem calcular matricialment el nombre de vegades que un terme o qualsevol dels seus refinaments era referit. Aquest fet es va recollir en forma d'una proposició i un corol.lari que van demostrar formalment.

En Renik va proposar una segona mesura que només depenia de la probabilitat d'un terme en comptes de l'IC. Basats en aquesta nova similaritat semàntica, vam provar que si la redefiníem en termes de distància, de fet es tractava d'una distància mètrica.

Les mesures d'estudi proposades per en Joslyn *et al.* són diametralment oposades a la d'en Lord *et al.* per quatre motius bàsics. Primer, és tracten d'aproximacions basades en l'estudi de les arestes. Segon, no són similaritats semàntiques, es tracten de "distàncies". Tercer, la mesura d'en Lord *et al.* es basa en un punt de vista probabilístic, mentre que les d'en Joslyn *et al.* es basen en un punt algebraic. I quart, les pseudo-distàncies només es poden calcular per a nodes comparables.

Joslyn *et al.* van proposar un formalisme diferent al d'en Carey per a definir el mapejat dels objectes en els termes de la ontologia, però no exempt de moltes semblances. Basat en el concepte Conjunts Parcialment Ordenats (POSET), Joslyn *et al.* introdueixen l'estructura de la Ontologia POSET (POSO) ([85]). Al respecte vam observar que un POSET és una estructura combinatòria bàsicament equivalent a un DAG. Aquest fet ens va fer notar que els conceptes usats en la definició d'un POSO poden ser fàcilment enllaçats amb les nocions de l'OOC. De fet, el POSET és el vocabulari de la ontologia, el conjunt d'objectes del POSO és l'objecte de l'OOC i la funció de mapejat és el mapejat entre la ontologia i la llista d'objectes en l'OOC. Per tant, ambdues estructures defineixen una manera de *com atribuir significat biològic*

Les pseudo-distàncies són mesures útils per calcular com de "diferents" són dos termes en POSET. No obstant, només tenen sentit quan es tracten de termes comparables. Ara bé, considerant la pseudo-distància de la longitud de la cadena mínima, vam demostrar que si ens centrem en el termes comparables, aleshores de fet es tracta d'una distància mètrica i vam apuntar que aquesta proba es podria estendre a les altres pseudo-distàncies.

El segon objectiu específic va ser desenvolupar un paquet `R` ([126]), anomenat `sims`, pensat per a calcular similaritats semàntiques entre els termes d'una ontologia arbitraria. Es van implementar 14 mesures d'aproximacions basades en l'estudi dels nodes i basades en l'estudi de les arestes. Tanmateix, es van implementar algunes funcions específiques per a treballar amb la GO. Les més notables permeten comparar dues llistes de gens basats en els perfils de similaritat semàntica, que vam definir com les llistes de similaritats semàntiques entre totes les paralles dels termes del graf induït per les dues llistes.

Durant el procés de desenvolupament del paquet, vam observar que algunes parts dels algorismes emprats en les funcions programades podien ser utilitzades per a demostrar matemàticament algunes propietats matemàtiques de la teoria de grafs com són la matriu simètrica associada a un graf simètric, el Teorema de Handshaking i el seu corol.lari ([155], [155], [17]).

Hi han altres paquets de `R` disponibles a la web del `Bioconductor` ([63]) per mesurar similaritats semàntiques entre els termes GO, concretament, el `GOSim` ([59]) i el `GOSemSim` ([165]). Ara be, primer, només poden calcular mesures de similaritat entre termes GO, en canvi el `sims` permet fer-ho per a qualsevol ontologia. Segon, les mesures de similaritat semàntica oferides per ell són moltes menys que les implementades al `sims`. Tercer, les seves similaritats semàntiques es centren només en l'aproximació basada en nodes. Quart, quan un usuari vol comparar dues llistes de gens o de termes GO ho pot fer aplicant funcions que calculen mesures de similaritat entre les llistes de termes i proporcionen un número que resumeix tota la informació. Des del nostre punt de vista, creiem que això fa perdre la visió de conjunt de les relacions entre les anotacions. Al respecte el `sims` ho soluciona proporcionant resums analítics i gràfics del perfil de similaritat semàntica, que permeten a l'usuari veure com són de similars o diferents aquestes relacions.

Les extensions naturals d'aquesta recerca es poden dividir en dues grans línies de treball. D'una banda, l'estudi que hem fet sobre la relació entre les mesures proposades i la distància mètrica es podria estendre a les aproximacions híbrides. Tanmateix, seria interessant intentar trobar una teoria que unifiqués les diferents aproximacions, i que permets canviar d'un tipus d'aproximació a l'altre. D'altra banda, hem observat que el `sims` té un comportament de càlcul molt interessant. Hem notat que calcula molt ràpidament les mesures de similaritat semàntica. Creiem que això pot ser donat perquè hem programat les funcions usant un enfoc matricial. En canvi, una inspecció ràpida de les funcions dels altres paquets semblen suggerir que han adoptat una posició més informàtica, basant-se en els bucles. No s'ha fet ni un estudi empíric, ni un estudi teòric basat en l'ordre de càlculs, per tal de provar que el `sims` és realment més ràpid. És doncs, evident, que seria interessant portar a terme aquesta comparació. Tanmateix, a diferència dels altres paquets, el `sims` no proporciona les mesures de similaritat semàntic normalitzades. En conseqüència, si es vol comparar els resultats de dues mesures, i fins i tot combinar-les no es podria fer en l'estat actual del paquet.

Per tant, una extensió en aquest sentit seria molt valuosa.

8.4 Classificació i estudi de les eines GO per a l'anàlisi d'enriquiment.

8.4.1 Material i mètodes

8.4.1.1 La selecció d'eines GO

Es va portar a terme una revisió d'una llarga llista d'eines disponibles a la pàgina web la GO ([148]). Degut a la gran heterogeneïtat entre els diferents tipus d'eines es va decidir centrar-se només en les *Eines per a l'anàlisi d'expressió/microarrasy* `http://www.geneontology.org/GO.tools.microarray.shtml`). Cal tindre present que la presència d'aquestes eines a la web de la GO ni garanteix l'aprovació del *The GO Cornsortium* ([152]), ni tampoc que s'hagin testat o que l'us de la informació sigui acurat. La taula 6.1 mostra la llista d'eines seleccionades, les entitats promotores, i les referències bibliogràfiques associades a aquestes.

8.4.1.2 La definició del Conjunt de Funcionalitats Estàndard i la classificació de les eines GO

Com a resultat de la revisió de les eines es va obtindre un alt nombre de característiques heterogènies. Després de diverses iteracions, les característiques seleccionades es van depurar per a ser convertirdes en una llista de 205 funcionalitats específiques estàndard.

Les capacitats de les eines avaluades es van classificar *in situ* d'acord amb el *Conjunt de Funcionalitats Estàndard* en base a tres criteris:

1. La funcionalitat estava disponible a la l'eina GO.
2. La funcionalitat havia estat mencionada a les referències, però no es va poder validar.
3. La funcionalitat ni es va trobar a la publicació ni a l'eina.

8.4.2 `SerbGO`: cerca de la millor eina GO

El `SerbGO` és un aplicatiu web dissenyat per a:

1. Facilitar als usuaris la tasca de determinar quina/es de les eines existents és/són les més apropiades per les seves necessitats.

2. Possibilitar una comparació entre algunes de les eines disponibles.

La figura 6.1 mostra el flux de treball del `SerbGO`.

El `SerbGO` és un aplicatiu web desenvolupat en `PHP` ([1]) utilitzant el suport de llibreries `ADOdb` per a `PHP` ([96]) i millorat amb el lleguatge `Javascript` ([108]) per a augmentar la seva interactivitat. L'aplicatiu funciona de manera acurada en la majoria de navegadors web, i ha estat testat, concretament, en els navegadors `Mozilla Firefox`, `Internet Explorer`, `Konqueror`, `Chromium` i `Opera`.

La informació sobre les eines està emmagatzemada en una base de dades relacional implementada amb el sistema `MySQL` ([76]).

8.4.2.1 Estudi de l'evolució i agrupament de les eines GO

La base de dades del `SerbGO` ha estat revisada periòdicament, però no regularment. Aquesta revisió ha consistit en eliminar eines que han deixat d'estar disponibles, actualitzar les millores fetes pels promotors, i afegir nous registres corresponents a la classificació de noves eines. Aquesta tasca ens va fer notar que hi havia un cert grau d'evolució en les eines classificades i emmagatzemades al `SerbGO`. Per tant, es va decidir fer un estudi estadístic basat en el seguiment de totes les eines incloses a la primera versió de la base de dades.

L'anàlisi de les dades està basat en el *nombre de funcionalitats estàndard* que tenien les 26 eines inicials. Es van fer tres talls transversals en el temps (2005, 2007 i 2009) (veure taula 6.1). Per aquestes eines es van descarregar sis taules (*tipus*, *espècies*, *dades*, *anotacions*, *estadístiques* i *sortides*) de la base de dades (veure secció 8.4.3.2), corresponents a cada any.

Les taules descarregades es van sotmetre a un procés d'homogeneïtzació de les dades amb l'objectiu d'eliminar redundàncies existents. Aquest procés va consistir en:

1. Homogeneizar alguns noms de camps de les taules de l'any 2005 amb els noms dels camps dels anys 2007 i 2009.
2. Reetiquetat dels valors d'algunes funcionalitats mencionades a les referències però no validades *situ.*
3. Reducció de les funcionalitats que eren massa específiques, les quals no aportaven informació extra i que podien afegir *soroll* a l'anàlisi.

El procés d'homogeneïtzació va deixar un total de 178 funcionalitats analitzables.

En base a les taules homogeneïtzades es van construir tres matrius de dades binàries, una per a cada any, on a les files hi havien les eines GO, a les columnes les funcionalitats, i a les cel.les hi havia un 1, si l'eina tenia una capacitat concreta o un 0 altrament.

L'anàlisi estadística es va dividir en tres parts complementàries: un estudi descriptiu, un anàlisi inferencial i un anàlisi multivariant. Tot l'anàlisi es va desenvolupar amb el programari estadístic `R` ([126]), el suport d'alguns paquets `R` extra, i algunes funcions programades explícitament. Aquestes funcions i tots els materials d'anàlisi es poden consultar al repositori del `GitHub` `https://github.com/jlmosquera/gotoolsevolution`.

Anàlisi descriptiu

Es va portar a terme per a obtenir un resum analític sobre les dades i va consistir en generar:

- *Taules de contingència* per a mostrar les freqüències de les funcionalitats estàndard per any, tant a nivell global com desagregades per grups de funcionalitats anomenats seccions.
- *Diagrames de barres* per a mostrar les distribucions de les freqüències per any.

Anàlisi inferencial

Es va fer per a testar si existien diferències entre les freqüències de les funcionalitats dels diferents anys, tant a nivell global com desagregades per seccions de funcionalitats. Concretament es va portar a terme el *Test de Khi-quadrat d'Homogeneïtat* ([115]) per a veure si les freqüències absolutes de les eines GO estaven distribuïdes idènticament a través dels anys. Aquests tests es van acompanyar de:

- *Boxplots* per a mostrar gràficament el nombre de funcionalitats de les eines GO per anys.
- *Scatterplots* per a mostrar gràficament les eines GO representades com a punts en un diagrama Cartesià a on cada eix estava associat a les freqüències de les funcionalitats d'un any específic.

- La *regressió local* (Loess) ([30], [31], [33]) per a proporcionar un resum gràfic de la relació entre les freqüències de funcionalitats disponible en les eines GO per a cada parell d'anys. Les corbes suavitzades i les bandes de confiança associades es van representar a cada scatterplot.

Anàlisi multivariant

Es va portar a terme per a explorar el comportament de les eines GO d'acord amb les seves capacitats al llarg del temps. Aquest anàlisi va consistir en:

- La construcció de dues *Matrius de dissimilaritats* per cada any, una, basada en el *coeficient de Jaccard* ([81]), i l'altre, basada en el *coeficient de Matching* ([143]).

- L'anàlisis de *Clusters jeràrquics aglomeratius* ([103], [52], [53], [73]) per a identificar grups d'eines GO per a cada any. Les mètriques de distàncies utilitzades es van basar en els coeficients de Jaccard i de Matching, i el mètode de clustering seleccionat va ser l'*Average Link* ([143]). Per determinar el nombre de clusters "òptim" es va portar a terme la representació dels:

  - *Dendrogrames* ([103], [52], [53], [73]) associats a cada cluster jeràrquic.
  - *Silhouette Plots* ([53], [87]) basats en el mètode no jeràrquic del *Partitioning Around Medoids*([53], [87]).

  I es van calcular els *Silhouette Coefficients* ([87]).

- El *Multidimensional scaling* (MDS) ([36], [18], [20], [18]) per a obtenir una representació espacial en dimensions reduïdes, de cada matriu de dissimilaritat associada a cada any, i ajudar amb la tasca d'identificació de *clusters* potencials. Per a cada matriu de dissimilaritats i any es va portar a terme dos MDS:

  - Un *MDS clàssic* ([153], [154], [65]). Per avaluar el grau de consens de la dimensionalitat es va usar la mesura P_m^2 de Mardia *et al.* ([103]) i es van representar diagrames de barres mostrant la "variablilitat explicada" per cada dimensió.
  - Un *MDS no-mètric de Kruskal* ([91], [138], [139]). Per a mesurar la pèrdua d'ajust es va calcular la mesura de *Stress-1* de Kruskal ([91]) i es van representar tant els *Scree plots* com els *diagrames de Shepard* ([18], [69]).

- El *Test de Mantel* ([101], [102], [142], [92], [93], [100]) per a estudiar l'associació entre cada parella de matrius de dissimilaritat, i el *Test Parcial de Mantel* ([142], [92], [93], [100]) per avaluar la correlació entre dues de les matrius condicionade per una tercera per tal de controlar efectes espuris.

8.4.2.2 DeGOT: Una ontologia per a desenvolupar eines GO

L'estudi de l'evolució de les eines GO va suggerir que la majoria de les necessitats associades a l'anàlisi d'enriquiment tradicional ja estaven cobertes. Tot i així, la comunitat científica ha seguit o be introduint millores a les eines existents, o be desenvolupant noves eines.

El desenvolupament d'un programari nou no és una tasca senzilla. Hi han molts factors i qüestions que cal tenir molt present durant el període de disseny del programa. Qüestions com:

- Per a quins tipus d'usuari va dirigida l'eina?
- A quines qüestions ha de donar resposta la informació emmagatzemada a la base de dades?
- Quines especies cobreix l'eina? O ha de ser independent de l'espècie?
- Quin tipus de mètodes estadístics inclourà?
- etc.

Els conceptes bàsics d'una ontologia

Les ontologies són la columna vertebral dels *Webs Semàntics* ([14], [74]). Hi han moltes maneres de definir i construir una ontologia ([67], [68]).

En general, les ontologies s'utilitzen per a capturar els conceptes d'un domini de coneixements amb la idea de facilitar la comunicació entre els usuaris i els computadors per a múltiples finalitats.

Un *domini de coneixements* és una forma de coneixement utilitzada per a fer referència a una àrea de l'esforç humà, una activitat computacional automatitzada, o qualsevol altra disciplina especialitzada ([77]) (p.e. els dominis de la GO 8.1.2).

La manera en que una *ontologia* intenta descriure els conceptes d'un domini, així com les relacions entre aquests conceptes, es fa a partir d'una descripció formal i explicitada. Aquestes formalitzacions són les *components de la informació* i són:

- Les *classes* d'objectes, que denoten els conceptes del domini.
- Les *propietats* que desciuen característiques i atributs de cada concepte (són les relacions).
- Les *facetes* que són restriccions sobre les propietats.
- Els *individus* que són els casos o exemples d'una classe. No totes les ontologies tenen aquesta component.

Implementació del `DeGOT`

Amb la idea de facilitar la tasca als desenvolupadors d'eines GO, s'ha construït una ontologia anomenada `DeGOT`, que pot donar resposta a algunes d'aquestes preguntes que s'han plantejat anteriorment 8.4.2.2.

`DeGOT` ha estat programada en el llenguatge `OWL` ([6]) usant el recurs `Protégé` ([117]).

8.4.3 Resultats

8.4.3.1 Definició del conjunt de funcionalitats estàndard

El conjunt de funcionalitats estàndard es va organitzar en 9 seccions (veure taula 8.1).

Seccions	Nombre de Funcionalitats
Tipus d'eina	2
Tipus d'experiment	7
Interfície	7
Disponibilitat	4
Espècies suportades	26
Dades	40
Anotacions	70
Anàlisi Estadístic	26
Sortida	23

Table 8.1: Nombre de funcionalitats estàndard definides per cada secció.

8.4.3.2 Classificació de les eines GO en base al conjunt de funcionalitats estàndard

En base al conjunt de les funcionalitats estàndard organitzat en nou seccions, es van construir sis taules que contenien la classificació de les 26 primeres eines GO avaluades (veure taules 7.8 a 7.13). Aquestes taules contenien la següent informació:

1. Taula amb els tipus d'eina, el tipus d'experiments/tecnologies, les interfícies i el tipus de disponibilitat.

2. Taula amb les espècies suportades.

3. Taula amb els tipus de dades i els identificadors d'entrada.

4. Taula sobre les fonts d'informació d'on cada eina es nodreix, el tipus d'anotacions funcionals que proporciona o suporta i les possibilitats que ofereix per a manegar les anotacions.

5. Taula amb els mètodes estadístics que implementats.

6. Taula associada als diferents tipus de sortides que ofereix cada eina GO.

8.4.3.3 Un aplicatiu web per a seleccionar i comparar eines GO (`SerbGO`)

El `SerbGO` és una eina web que està lliurament disponible i no requereix un *login.* Es pot accedir directament al servidor (`http://estbioinfo.stat.ub.es/apli/serbgo`) del *Grup de Recerca Estadística i Bioinformàtica* liderat pel Dr. Alex Sánchez. Tanmateix l'eina es va sotmetre i vas ser acceptada per estar disponible a la web del consorci de la GO (`http://www.geneontology.org/GO.tools.microarray.shtml#serbgo`). La figura 7.1 mostra la imatge de benvinguda del `SerbGO`.

L'eina consisteix en una sèrie de formularis que permeten accedir a la informació classificada i emmagatzemada en una base de dades relacional. Aquesta base de dades conté set taules. Sis d'aquestes taules són les esmentades a la secció anterior 8.4.3.2 i la setena és la taula principal que conté: els noms de les eines, els promotors de les eines, les referències consultades i diferents índex interns, que faciliten l'accés a les altres taules quan es fan les consultes via els formularis web.

Fluxe de treball de l'aplicatiu

El `SerbGO` ofereix dues possibilitats d'execució:

1. Seleccionar una llista de capacitats del conjunt de funcionalitats estàndard (figures 7.2 a 7.7), per a cercar les eines GO que satisfan aquestes característiques .
2. Seleccionar una llista d'eines GO (figura 7.8) per a ser comparades.

En el primer cas, la cerca dóna lloc a un taula amb dues columnes (figura 7.8) a on a la primera columna hi ha el nom de les eines GO que compleixen els requisits i a la segona columna les dades del promotor de l'eina. Els noms de les eines són enllaços directes a les adreces corresponents. Al final de la taula, hi han un botó (*Find*) que en ser clicat mostra una nova taula a on a les files hi ha les funcionalitats, a les columnes els noms de les eines (també enllaçades als llocs web respectius) i a les cel.les està indicat si una eina disposa o no d'una capacitat específica (figura 7.10). En el segon cas, es genera una taula com la segona que acabem de descriure, però en comptes de mostrar la comparació resultant de la cerca del robot del `SerbGO`, hi han les capacitats de les eines GO seleccionades.

Període de probes de l'eina

El període de probes de la versió beta del `SerbGO` es va portar a terme considerant només les 26 eines classificades 6.1. En aquest procés hi van participar persones de diferents centres i organitzacions arreu del món amb les que es va anar contactant a mesura que s'anava presentant l'eina a diferents esdeveniments. També hi van participar alguns desenvolupadors d'aquestes eines (p.e. com els del `FatiGO` ([3]), el `GARBAN` ([105]) o el `BiNGO` ([99]), qui van suggerir algunes millores que es van incorporar i validar *a posteriori*.

El `SerbGO` ha estat funcionant des del Juny de 2006 i l'article associat va ser publicat a l'any 2008 a la revista indexada *Nucleic Acids Research*. L'eina ha estat actualitzada en diverses ocasions, però no periòdicament per motius de recursos. Al final d'aquesta tesis, la base de dades del `SerbGO` conté emmagatzemada la classificació de 50 eines.

8.4.3.4 Evolució i clustering de les eines GO

L'anàlisi global descriptiu suggereix que hi va haver un augment del nombre de capacitats que oferien les eines al llarg del temps. I sembla indicar que el canvi més substancial es va produir a l'any 2007 (taula 7.14). Aquest canvi es pot apreciar clarament al diagrama de barres (figura 7.11).

A la taula 7.15 es pot veure el nombre de funcionalitats per secció. Les freqüències absolutes i relatives de les funcionalitats disponibles per secció associades a cada eina GO es poden consultar a les taules 7.16, 7.17 i 7.17. I els diagrames de barres associats a cadascuna d'elles es poden veure a la figura 7.12.

Es fàcil veure en aquestes representacions que les funcionalitats per seccions han experimentat un augment en el nombre de funcionalitats proporcionat per les eines GO al llarg del temps. Sembla però, que els promotors de les eines han invertit més esforços en les capacitats d'anotacions i el nombre d'espècies suportades. Tanmateix, sembla que l'augment produït en les funcionalitats de les altres seccions sigui més "homogeni".

A nivell global, l'anàlisi inferencial mostra que considerant un nivell de significació del 0.05, els canvis observats descriptivament són significatius entre els anys 2005 i 2007, i entre els anys 2005 i 2009 (taula 7.19). És a dir, rebutgem la hipòtesi nul.la de que la distribució de les freqüències entre les parelles d'anys indicats és la mateixa, però no tenim suficient evidència com per rebutjar la hipòtesi nul.la associada a la comparació 2007 *vs*2009. En altres paraules, a nivell global, els tests d'homogeneïtat de la Khi-quadrat ens està dient que l'augment observat en el nombre de funcionalitats, és significatiu entre els anys 2005 i 2007, i els anys 2005 i 2009, però no podem assegurar que l'augment observat entre el 2007 i el 2009 sigui significatiu.

Comparison	Chi.Square	df	PValue	Adj.Pvalue
2005 *vs.* 2007	95.29502	25	3.842300e-10	5.763450e-10
2007 *vs.* 2009	33.87294	25	1.106518e-01	1.106518e-01
2005 *vs.* 2009	100.18731	25	5.834561e-11	1.750368e-10

Table 8.2: Resultats dels test de Khi-quadrat d'homogeneïtat entre les distribucions de les freqüències de les funcionalitat per a cada parella d'anys contrastada.

L'augment significatiu en el nombre de funcionalitats proporcionades per les eines també es pot notar les representacions gràfiques dels boxplots i els scatterplots (figura 7.13).

En canvi, quan desagreguem per seccions es pot observar que només són significatius aquests canvis entre els anys 2005 *vs* 2007 i 2005 *vs* 2009 en la secció d'espècies suportades i totes les comparacions associades a la secció d'anotacions (taula 7.20). Observat les figures associades (7.14 to 7.19) es pot apreciar una clara tendència cap amunt en els boxplots al llarg del temps. En el cas dels scatterplots, les figures associades al 2007 *vs* 2009 semblen mostrar un creixement "lineal", no essent el cas dels anys 2005 *vs* 2007 i 2005 *vs* 2009, on els núvols de punts tenen una forma que tendeix a ser més circular, suggerint un aument en el nombre de funcionalitats no tant homogeni.

En la selecció del nombre de clusters, basant-nos en els valors dels *Silhouette Coefficients* (SC), s'observa una pèrdua substancial de les estructures dels clusters, en ser aquests inferior al 0.25. Això suggereix que els models d'eines GO que es determinin via clusters basats en aquestes matrius de dissimilaritats podrien no estar ben caracteritzats (taula 8.3).

Year	Jaccard Coefficient		Matching Coefficient	
	Num. Clusters	SC	Num. Clusters	SC
2005	2	0.13	3	0.2
2007	9	0.09	16	0.1
2009	2	0.1	5	0.12

Table 8.3: Taula resum del nombre òptim de clusters d'acord amb els calors SC basats en els coefficients de Jaccard i de Matching.

Tanmateix, depenent de l'any i del tipus de coeficient el nombre de clusters és variable. Aquest fet, de difícil interpretació, podria estar associat en algun sentit amb les les millores introduïdes a les eines entre els anys 2005 i 2007.

Observant els diagrames de barra dels *average silhouette widths* (figures 7.20 i 7.21), independentment de l'any, del coeficient usat i dels valors SC, sembla que un nombre baix de clusters poden explicar una gran part de la informació sobre els grups d'eines GO. Per tant, basats en aquest fet i en ordre a discutir l'evolució de les funcionalitats de les eines, s'ha decidit considerar 3 clusters que s'han destacat tant en els dendrogrames associats als cluster jerarquics com en les representacions dels MDS.

Globalment parlant, els clusters jeràrquics mostren un la gran nombre d'eines que cauen en un grup majoritari i la resta d'eines es reparteixen en dos grups menors (figures 7.22, 7.23 i 7.24). Aquest fet és molt notable quan observem els dendrogrames associats al coefficient de Jaccard, on a l'any 2009 el cluster majoritari aglutina quasi totes les eines. Notis, que aquest comportament d'"homogeneïtzació" està va en consonància amb la pèrdua d'estructura suggerida anteriorment pels valors de SC. En el cas del coeficient de Matching, el comportament és diferent. Tot i mantenir-se la idea global comentada abans, les representacions dels clusters jeràrquics semblen suggerir que les eines del grup majoritari es van repartir en els grups minoritaris. És a dir, semblen "especialitzar-se". Aquest fet *a priori* contradictori es podria atribuir a la definició de les fórmules dels coeficients de Jaccard i de Matching, on el primer, només té en compte quant dos eines tenen una capacitat en comú, i el segon també considera quan no tenen una capacitat en comú. En qualsevol cas, deixant de banda els valors de SC, en el coeficient de Matching sembla haver-hi un subconjunt d'eines GO que sempre semblen "anar" juntes i són: el `CLENCH` [137], l'`ermineJ`, el `FuncAssociate` [16], el `GOArray`, el `GoSurfer` [169], l'`OntoGate (OntoBlast)` [167], l'`ontology Traverser` [164], i el `SeqExpress` [19]. Notis que aquestes eines són els punts dels scatterplots, de la figura 7.13, que cauen aproximadament a la bisectriu imaginaria, és a dir, les eines que semblen no haver experimentat grans canvis.

Les representacions en dues dimensions de les solucions dels MDS es mostren a les figures 7.22, 7.23, i 7.24. Un inspecció general a través del temps suggereix que les distàncies entre els punts són lleugerament més grans en les solucions basades en el coeficient de Jaccard que en les del coeficient de Matching. Tanmateix, independentment del coeficient, no s'observa un efecte de separació evident entre grups d'eines, però si es fa notar que els punts de les solucions no mètriques tenen un comportament més homogeni, que els punts de les solucions clàssiques. Ara be, els núvols de punts mostren un comportament molt subtil i curiós. Les representacions dels MDS a l'any 2007 mostren una tènue contracció del gran núvol de punts, per després experimentar una lleugera expansió de les distàncies entre els punts, més emfatitzada en la primera dimensió. Aquest fet no és de fàcil d'"interpretació", però creiem que podria estar lligat a algun efecte associat amb les millores introduïdes pel desenvolupadors que hem observat tant a la descriptiva com en e les comparacions significatives.

En xifres, els percentatges de les mesures de consens d'*adequacy* (figura 7.25) associades a les solucions mètriques basades en el coeficient de Jaccard són 61.23%, 52.67% i 53.64%, i els associades al coeficient de Matching són 63.39%, 73.93% i 71.63%. És a dir, en el primer cas la variabilitat explicada suggereix que la representació en dues dimensions no és bona, però tampoc és dolenta. En canvi en el segon cas, les variabilitats explicades són bastant més bones. Les representacions de les solucions no mètriques mostren uns valors de Stress associats al coeficient de Jaccard del 18.48%, 16.74% i 15.63% , i els valors associats al coeficient de Matching són del 17.98%, 17.14% i 20.94%. Per tant, en base als criteris suggerits per en Kruskal (taula 6.4) ([91]), la bondat d'ajustament en ambdós casos es entre feble i pobre.

Els valors de Stress ha estat àmpliament criticats per ser sobre-simplistes o ser molt influenciables per *outliers* ([160], [18]) Per aquest motiu, aquests valors s'han acompanyat de les representacions dels Scree plots i dels diagrames de Shepard (figures 7.26 i 7.27). Resumint, els Scree plots suggereixen que les representacions en dues dimensiones no són suficientment fiables i per tant és difícil identificar el clusters d'eines GO. D'altra banda, els diagrames de Shepard suggereixen que les distàncies i les disparitats entre les eines GO a les representacions dels MDS no mètrics aproximen be les proximitats originals. Ara bé, cal tenir present que els punts del diagrama de Shepard no són estrictament projeccions de les proximitats, si no una projecció de les matrius de dissimilaritat en unes dimensions reduïdes. Així doncs, combinant els resultats dels Scree plot i els diagrames de Shepard, suggereixen que caldria augmentar la dimensionalitat de la configuració dels MDS no mètrics (a aproximadament 7) per a explicar la informació de forma més fiable. En conseqüència, els "models" potencials o especialitzacions de grups d'eines observats *a priori* no són del tot creïbles.

La taula 8.4 mostra els resultats dels tests de Mantel, simples i parcials. Considerant un nivell de significació del 0.05, en el test simple de Mantel, les correlacions entre cada parell de matrius de dissimilaritats, independentment del coeficient considerat, són estadísticament significatives. Però, els coeficients no són suficientment alts, amb l'excepció de la correlació entre les matrius associades al coeficient de Jaccard entre els anys 2007 i 2009. Pel que fa als tests de Mantel parcials, no tenim suficient evidència estadístic com per a rebutjar la hipòtesi nul.la. És a dir, independentment del coeficient, quan estudiem la correlació entre les matrius de dissimilaritats del 2005 i del 2009 controlant l'efecte del 2007, no s'observa una relació "lineal". En altres paraules, les similaritats entre les eines GO no semblen ser les mateix al llarg del temps

8.4.3.5 Una ontologia per a desenvolupar eines GO (`DeGOT`)

`DeGOT` és una ontologia simple dirigida a proporcionar als desenvolupadors d'eines GO un vocabulari estructurat que els ajudi a dissenyar una nova eina

(Partial) Mantel Test	r_M	PValue
$r_M(D^J_{2005}, D^J_{2007})$	0.450	0.0001
$r_M(D^J_{2007}, D^J_{2009})$	0.803	0.0001
$r_M(D^J_{2005}, D^J_{2009})$	0.276	0.0033
$r_M(D^J_{2005}, D^J_{2009} \vert D^J_{2007})$	-0.160	0.9727
$r_M(D^M_{2005}, D^M_{2007})$	0.479	0.0002
$r_M(D^M_{2007}, D^M_{2009})$	0.588	0.0001
$r_M(D^M_{2005}, D^M_{2009})$	0.283	0.0001
$r_M(D^M_{2005}, D^M_{2009} \vert D^M_{2007})$	0.003	0.4863

Table 8.4: Mantel and Partial Mantel Tests.

or a introduir millores en una eina ja existent. Aquesta ontologia està lliurement disponible al servidor (`http://estbioinfo.stat.ub.es/apli/degot`) del *Grup de Recerca Estadística i Bioinformàtica* liderat pel Dr. Alex Sánchez.

En el web del `DeGOT` es pot consultar la documentació (generada amb `LODE` ([121])) i la descripció de tots els elements de la ontologia i descarregar-se el codi de la ontologia. Al respecte, per a navegar i explorar les anotacions del `DeGOT` és necessari utilitzar un navegador d'ontologies `OWL`. Es recomana utilitzar el programari `Protégé` descarregar-se i instal.lar-se del web `http://protege.stanford.edu/`.

Domini de coneixement del `DeGOT`

El domini de coneixement del `DeGOT`està centrat en les característiques de les eines GO. Els termes de la ontologia permeten compartir i reutilitzar coneixement comú de les estructures de les funcionalitats entre els usuaris i els desenvolupadors, fer assumpcions explícites sobre el domini, separar el coneixement del domini del coneixement funcional o operatiu de les eines GO i explotar el domini de coneixement per tal de ser usat com a complement de cerques i comparacions am el `SerbGO`.

La organització jeràrquica del `DeGOT` permet fer actualitzar i afegir característiques de manera molt fàcil i ràpida.

Els constructors de la ontologia

Actualment, el `DeGOT` està format per 314 classes, 18 propietats i 4 individus.

Les classes El node arrel de la ontologia es diu *GOTool_Domain_Concept*[1]. Aquesta classe té quatre fills que es van especialitzant a mesura que aprofundim en la ontologia. Aquestes subclasses són: *Availability* (tipus d'interficies, llicències, sistemes operatius de de leina GO), *File_Format* (formats dels arxius que utilitza o proporciona l'eina), *Functionality* (tipus d'entrades, anàlisis i sortides permeses per l'eina GO) i *Resource* (bases de dades, eines i altres recursos que estan associats am l'eina).

Les propietats Les propietats del DeGOT són relacions binàries entre els individus. La taula 7.25 proporciona la llista de les propietats implementades a la ontologia, juntament amb el seu domini (origen) i rang (termini), i les propietats inverses respectives.

No s'han definit restriccions sobre les propietats del DeGOT.

Els individus Els individus anotats a la ontologia són noms d'eines GO existents. Els individus poden pertànyer a més d'una classe. En el DeGOT s'ha anotat quatre individus (agriGO ([49]), BinGO ([99]), CateGOrizer ([78]), and CLENCH ([137])).

8.4.4 Discussió

La segona part d'aquesta tesi estava dirigida a respondre els sis objectius específics associats amb l'estudi de les eines GO per a l'anàlisi d'enriquiment.

Donada la gran quantitat d'eines GO per a l'anàlisi d'enriquiment ([47], [79]), el primer dels objectius d'aquesta tesi va ser definir un *Conjunt de Funcionalitats Estàndard* que permetes classificar les eines GO en base a les seves capacitats. Així doncs, per a crear aquest conjunt de funcionalitats es van examinar una llarga llista de referències literàries associades a eines GO, disponible al web del *The GO Consortium* ([152]). Fruit d'aquesta revisió es va construir una llista amb 205 característiques organitzades en 9 seccions diferents, i en base a aquest Conjunt de Funcionalitats Estàndard es van classificar 26 eines GO. Aquesta classificació va donar lloc a la construcció d'unes taules que recollien les capacitats que proporcionaven cadascuna de les eines.

[1] A les ontologies OWL, i en general totes, el node arrel s'anomena *Thing*, però per interès propi en la descripció del DeGOT, hem decidit descriure com a node arrel el seu únic fill, *GOTool_Domain_Concept*.

Amb l'objectiu d'explotar tota aquesta informació es va decidir desenvolupar un aplicatiu web anomenat SerbGO [111] dissenyada per a ajudar als usuaris a seleccionar i comparar les eines GO i així puguin trobar quines d'elles són les que millor s'adapten als seu objectius.

La base de dades del SerbGO ha anat sent actualitzada periòdicament des de la seva versió. Aquest fet va fer-nos notar que a banda de les noves eines classificades segons el Conjunt de Funcionalitats Estàndard, molts promotors introduïen noves capacitats en les eines existents o milloraven algunes de les seves capacitats. En base a aquest monitoratge, ens vam preguntar si s'estaria produint una especialització de les eines existent, o si hi hauria una certa redundància en les capacitats de les eines i en conseqüència s'estarien invertint esforços i recursos en donar resposta a qüestions que ja estaven solucionades. Per tant, vam decidir de fer un anàlisi estadístic amb l'objectiu d'estudiar l'evolució i *clustering*, si és que hi era, de les primeres 26 eines GO classificades al SerbGO.

L'anàlisi estadístic es va plantejar en tres parts: una descriptiva, una anàlisi diferencial i una anàlisi multivariant, pensats per a descriure els percentatges de les capacitats oferides per les eines, observar si aquests hi havien diferències significatives, tant a nivell global com per (seccions (o grups) de funcionalitats, i veure si les similaritats entre les eines evolucionava al llarg del temps permetent identificar models d'eines GO. Els resultats de les tres l'anàlisi suggerien que efectivament els promotors havien invertit esforços en augmentar el nombre de capacitats de les eines, i van ser significativament diferents en les capacitats d'anotació i d'espècies suportades per les eines. Tanmateix, no es va poder identificar amb claredat agrupacions d'eines degut a una pèrdua d'estructura de clusters. Tanmateix, l'exploració multivariant sembla suggerir que a mesura que pasa el temps les eines van homogeneïtzant les seves capacitats.

A la vista monitoratge de les eines, dels resultats obtinguts i del fet de què la comunitat científica segueix invertint esforços en millorar les eines existents o construir-ne de noves, ens vam plantejar la idea de crear un recurs que donés suport als desenvolupadors quan es troben amb la tasca de dissenyar una nova eina GO. Aquesta reflexió va ser materialitzada amb el desenvolupament d'una ontologia, que vam anomenar DeGOT. El DeGOT és una ontologia que proporciona un vocabulari sobre el coneixement de les funcionalitats de les eines GO, i que està pensada tant per a ajudar als desenvolupadors quan es troben amb la tasca dissenyar una nova eina, com per a ser utilitzada com a suport de cerques fetes amb el SerbGO. Aquesta ontologia està formada per 314 classes, 18 propietats d'objectes i 4 individus, i fàcilment pot ser estesa a noves característiques.

Durant els darrers anys el creixement d'estudis d'integració de dades òmiques i/o la combinació de mètodes per a millorar el coneixement biològic és ha experimentat un canvi notable a l'ala. En aquest sentit, una de les extensions més interessants que es podria portar a terme en futures línies de recerca, seria investigar la possibilitat de combinar diferents mètodes i eines per tal de proporcionar un anàlisi d'enriquiment molt més plausible i/o informatiu. Tanmateix, tot i que el procés de classificació i posterior seguiment és una tasca d'envergadura considerable en termes de recursos i de temps, seria interessant construir una eina com el SerbGO que ens permetés classificar a la nova moda d'eines per a fer anàlisis d'enriquiment, que es basa en l'anàlisi de xarxes biològiques.

8.5 Conclusions

Aquesta tesi s'ha centrat en els mètodes i eines per assignar interpretació biològica basada en la *Gene Ontology* a dades generades en experiments omics. La recerca ha explorat dos aspectes principals:

1. L'estudi de dos tipus de mesures de similaritat semàntica per explorar les categories de la GO.

2. La classificació i estudi de les eines GO per a l'anàlisi d'enriquiment.

- Respecte al primer punt:

    1. S'ha demostrat que:
        (a) La matriu d'accessibilitat associada a un graf simètric, és simètrica.
        (b) El *Teorema de Handshaking* i el seu corol.lari poden ser demostrats en base a la matriu d'incidències.
        (c) La propietat monòtona de la probabilitat pot ser verificada en termes del formalisme proposat per en Carey.
        (d) El node arrel d'una ontologia és el terme amb el Contingut d'Informació més baix, que de fet és nul.
        (e) Per a calcular el *Contingut d'Informació*, es pot utilitzar el producte matricial de la matriu dels nombres de camins de qualsevol longitud entre cada parell de termes per la matriu de mapejat i obtindre el nombre de vegades que cada terme o qualsevol de les seves especialitzacions apareix a la ontologia.
        (f) La segona mesura de Resnik redefinida en termes de distància és una distància mètrica.
        (g) Quan ens restringim a termes comparables, la pseudo-distància de la longitud de la cadena mínima és una distància mètrica.
    2. S'ha vist que:
        (a) Hi ha un cert nivell d'analogia entre el concepte de l'*Object-Ontology Complex* i la *Partially Ordered Sets Ontology.*
        (b) La mesura de Lord *et al.* és la de fet la mesura per Resnik.
    3. S'ha desenvolupat un paquet `R` anomenat `sims` que:
        (a) Està dirigit per a calcular similaritats semàntiques.

   (b) S'ha implementat un gran nombre de mesures de dues aproximacions diferents.

   (c) Proporciona un punt de vista alternatiu per a comparar dues llistes de gens basat en els perfils de similaritat semàntica.

   (d) Està disponible lliurement al repositori de `GitHub` `https://github.com/jlmosquera/sims`.

- Respecte al segon punt:

   1. S'ha vist que la definició d'un *Conjunt de Funcionalitats Estàndard* permet classificar les eines GO per anàlisi d'enriquiment.

   2. S'ha desenvolupat una eina web anomenada `SerbGO` que:

      (a) Està dirigida per a seleccionar i comparar eines GO per l'anàlisi d'enriquiment.

      (b) Està disponible lliurement al servidor del *Grup de Recerca en Estadística i Bioinformàtica de la UB* (`http://estbioinfo.stat.ub.es/apli/serbgo`) i al web del *The GO Consortium* (`http://www.geneontology.org/GO.tools.microarray.shtml#serbgo`).

   3. L'estudi de eines GO ha revelat que:

      (a) Els promotors han introduït millores en el eines GO per a l'anàlisi d'enriquiment al llarg del temps.

      (b) Les eines GO han evolucionat homogèniament i no s'han trobat grups ben definits d'eines GO.

   4. S'ha desenvolupat una ontologia anomenada `DeGOT` que:

      (a) Proveeix un vocabulari organitzat per ajudar a desenvolupadors quan necessiten disenyar una nova eina GO o millorar una ja existent.

      (b) Pot ser utilitzada per a suportar cerques i comparacions d'eines GO portades a terme amb el `SerbGO`.

      (c) Està disponible lliurement al servidor del *Grup de Recerca en Estadística i Bioinformàtica de la UB* (http://estbioinfo.stat.ub.es/apli/degot)

Part VI

Appendices

Appendix A

Main Documents Attached to the Thesis

A.1 Papers

1. Alex Sánchez and **Jose Luis Mosquera**. *The quest for biological significance.* In Luis L. Bonilla, Miguel Moscoso, Gloria Platero, and Jose M. Vega, editors, Progress in Industrial Mathematics at ECMI 2006, volume 12 of Mathematics in Industry, pages 566570. Springer Berlin Heidelberg, 2008.

2. **Jose Luis Mosquera** and Alex Sánchez-Pla. *SerbGO: searching for the best GO tool.* Nucleic Acids Research, 36(Web Server issue):W368-371, 2008.

A.2 Software

1. `sims` vignette: it shows the management and main possibilities of the package through different examples.

A.3 Other Papers

Papers in statistics and bioinformatics not related to the thesis and published in international per reviewed journals during the perios of the thesis. Just the first page is attached.

1. Sara Fernández, *Jose Luis Mosquera*, Lide Alaña, Alex Sánchez-Pla, Juan Morote, Santiago Ramón y Cajal, Jaume Reventós, Inés de Torres and Rosanna Paciucci *PTOV1 is overexpressed in human high-grade malignant tumors.* Virchows Archive, 458(3):323-330, 2011.

2. Cristina Martínez, Marıa Vicario, Laura Ramos, Beatriz Lobo, **Jose Luis Mosquera**, Carmen Alonso, Alex Sánchez, Mar Guilarte, María Antolín, Inés de Torres,Ana M. González-Castro, Marc Pigrau, Esteban Saperas, Fernando Azpiroz and Javier Santos. *The Jejunum of Diarrhea-Predominant Irritable Bowel Syndrome Shows Molecular Alterations in the Tight Junction Signaling Pathway That Are Associated With Mucosal Pathobiology and Clinical Manifestations.* American Journal Gastroenteroly, 107:736-746, 2012.

3. Gisela Nogales-Gadea, Alba Ramos-Fransi, Xavier Suárez-Calvet, Miquel Navas, Ricard Rojas-García, **Jose Luis Mosquera**, Jordi Díaz-Manera, Luis Querol, Eduard Gallardo and Isabel Illa mail *Analysis of Serum miRNA Profiles of Myasthenia Gravis Patients.* PLoS ONE 9(3):e91927, 2014.

The Quest for Biological Significance

Alex Sánchez[1] and Josep Lluis Mosquera[1]

Departament d'Estadística. Universitat de Barcelona. Facultat de Biologia. Avda Diagonal 645. 08028 Barcelona. Spain. asanchez@ub.edu;jlmosquera@ub.edu

1 Introduction

With the advent of genomic technologies it has become possible to perform, in a routinely manner, new types of experiments to analyze simultaneously the behavior of thousands of genes or proteins in different conditions. A common trait in these type of studies is the fact that they generate huge quantities of data what has lead to using the term "high-throughput" to describe them. There are different types of high-throughput experiments, but we will refer from now on to the most well known ones: microarray experiments.

A typical microarray experiment is one who looks for genes *differentially expressed* between two or more conditions. That is, genes which behave differently in one condition, for instance healthy or untreated cells, than in another, for instance tumor or treated cells. Such an experiment will result very often in long lists of genes which have been selected using some criteria, such as a t-test, to assign them *statistical significance*.

Most of the times the biological interpretation of the list is not obvious. Sometimes the number of items selected as being statistical significant is very high and it seems reasonable to (try to) synthesize them looking at *what the list means from the biological point of view*. Sometimes, instead, the selected items do not show any statistical significance, but even so, it is expected -or it seems clear- that, biologically, they "mean something", probably related to the process being analyzed.

In whatever of the previous situations we find, the usual way to proceed is to shift the focus from "statistical" to "biological" significance. There is a clear agreement about what does statistical significance mean. However there is no consensus definition of biological significance at all. Although everyone talks about it...

1.1 So, what does Biological Significance mean?

Interestingly what many authors do to define biological significance is to redefine it in terms of statistical significance. This can be clearly seen in [1] who describes it as:

> ... *to understand the biological relevance of statistical differences in gene expression data* by examining significant differences in the distribution of (GO) terms related to biological processes or molecular function.

This is not however the only possible definition. For instance `GeneSifter` (`http://www.genesifter.net/web/`) a company presenting their goals as to " ... make it easier to understand the biological significance of your microarray data" does not give any definition of the term. The nearest explanation of what they mean by this is the following:

> ... *to characterize the biology involved in a particular experiment*, and to identify particular genes of interest ... combining the identification of broad biological themes with the ability to focus on a particular gene ...

In any case, it is clear that whatever they mean by Biological Significance they do not relate this to Statistical Significance.

In short. Establishing the biological significance of high throughput experiments is an important step for their success and many efforts are addressed to this. Less efforts, it seems, than to clarifying what the term exactly means.

1.2 The Gene Ontology

Attempts to perform a biological interpretation of high throughput experiments are often based on the Gene Ontology (GO), an annotation database created and maintained by a public consortium, the Gene Ontology Consortium[1], whose main goal is, citing their mission, *to produce a controlled vocabulary that can be applied to all organisms even as knowledge of gene and protein roles in cells is accumulating and changing*. The GO is organized around three principles or basic ontologies: (1) Molecular function (MF), which describes tasks performed by individual gene products;(2) Biological process (BP), which describes broad biological goals, such as mitosis (cell division) and (3) Cellular component (CC) describing subcellular structures, locations, and macromolecular complexes such as nucleus, or other organelles. A given gene product may represent one or more molecular functions, be used in one or more biological processes and appear in one or more cellular components. Each ontology (MF, BP or CC) consists of a high number of terms or categories hierarchically related from least (top) to most (bottom) specialized

[1] www.geneontology.org

characteristics. Ontologies are indeed direct acyclic graphs (DAG) and graph theory is clearly one possible, although not yet generalized, approach for their study. Most genes are annotated in one or more categories. Annotations are made as specific as possible. As a consequence a gene is associated not only with its annotations but also with all the less specific terms associated with them. This altogether configures a network of terms for each gene integrated in the bigger network which is the GO (see figure

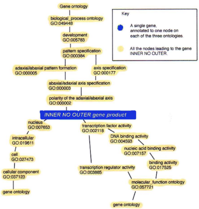

Gene ontology
biological_process ontology GO:049448
development GO:005783
pattern specification GO:000384
adaxial/abaxial pattern formation GO:000005
axis specification GO:000177
abaxial/adaxial axis specification GO:000003
polarity of the adaxial/abaxial axis GO:000002
INNER NO OUTER gene product
nucleus GO:007653
intracellular GO:019611
cell GO:027473
cellular component GO:037123
gene ontology
transcription factor activity GO:002118
DNA binding activity GO:004593
nucleic acid binding activity GO:007157
binding activity GO:017525
transcription regulator activity GO:003885
molecular_function ontology GO:057721
gene ontology
Key
A single gene, annotated to one node on each of the three ontologies.
All the nodes leading to the gene INNER NO OUTER.

Fig. 1. A hypothetical example of GO annotations for the gene “INNER NO OUTER”. Every gene is annotated in the three ontologies, MF, BP and CC.

2 From Biological to Statistical Significance: Gene Enrichment Analysis

In recent years there have been developed many methods intended to quantify Biological Significance (*BS* from now on) in terms of Statistical Significance (*SS* from now on). Draghici et al. ([2]) consider as many as 15 related applications which perform in different but related ways. In this paper we will not even attempt to compare or offer a panoramic view of the existing methods, although in the appendix we describe a tool that we have developed precisely with this goal in mind. Instead we will center on what is possibly the most well-known and most used approach to obtaining *BS* from *SS*.

Assume that we have the results of a typical microarray experiment where we have selected K “interesting” genes from a wider population or Universe, of size N. Each gene is annotated to one or more GO categories so that we end up with a subset $\{A_1, A_2, ...A_G\}$ of annotated categories. Gene Enrichment Analysis (GEA) consists of performing a statistical test *separately for each category* A_i, $i = 1, ...G$ to decide if the proportion of genes in the sample which have been annotated in category A_i is the same as those in the Universe belonging to the same category. If this is so one can interpret that this category is not related to the biological phenomenon that led to select the genes in the sample. Oppositely if the proportion of genes in the sample appearing in category A_i is greater (enriched) or smaller (impoverished) that those in the Universe one can assume that this category is *Biologically Significant*. GEA can easily be formulated in terms of hypergeometric sampling allowing to use the hypergeometric distribution to compute p–values for the test having null hypothesis: H_0 *The GO category* A_i *is equally represented in the Universe than in the group of differentially regulated genes*. Details of this test can be found for instance in [2]

3 Discussion: Drawbacks and limitations

Keeping in mind that, for the sake of centering on the difference between *SS* and *BS*, we have adopted a simplified view it is clear that the previous approach shows some limitations.

By one side, and this is applicable mainly to GEA, the method selects categories separately, without explicitly caring for relations between them. This, jointly with the fact that it relies on a statistical filtering criteria, suggests that is useful to highlight biologically relevant "hot spots" but it does not offer a global picture of what is happening in the biological side of the experiment.

Instead of looking at more and more methods checking their virtues and defaults (but see the appendix and [4]) it is good perhaps to remark another important flaw: When we rely in *SS* to define *BS* we depend on p–values at one or two levels, that is those p–values that have been used to select the genes, and those p–values computed to check the significance of the categories. However p–values are not free from criticisms (see [3]). They depend on underlying probability models and are often subject to misinterpretation as well as used to justify otherwise unjustifiable cutoffs. In short using p–values to define *BS* we risk to translate into it the abuse that has sometimes been observed with its use to define *SS*.

3.1 Towards a new definition of Biological significance

Our goal in the previous lines has been mainly to emphasize that simply relying on statistical significance to define biological significance can be as misleading as just using but not defining the term. And the interesting point is precisely this: biological significance is not an entelechy. An expert in a given biological field will often be able to distinguish between two sets of results and chose those that can be considered more relevant. The challenge for mathematicians, statisticians and other scientists working in parallel with those experts is to develop an approach which tells a story which is, at the same time, as objective as possible, but also as near to the biologist's choice as can be obtained. Probably it will requires approaches that integrate information from several sources and are able to combine weak non significant evidences with more objective results into relevant conclusions that can be considered biologically significant, not because somewhere a p-value is tiny, but because they really mean something.

Appendix

In this work we have explicitly avoided making comparisons between the existing methods or tools. It is not an easy task because there exists dozens of them and they are not free of redundance at all.

This is in itself a barrier for a potential user because even if she understands clearly what she is looking for she will be faced to choose between many similar tools.

To help users in this decision process we have developed `SerbGO` (for **Se**a**r**ching the **b**est **GO** tool). It is a free web based tool that can be used in any two directions: One can ask for the desired functionalities and find out which are the programs that include them or one can simultaneously analyze several tools to find out which functionalities are implemented and which are missing.

SerbGO is available at `http://estbioinfo.stat.ub.es/apli/serbgo/`.

The program has proven useful not only to users who wish to find the tool they need or who want to compare several tools. It has helped also to classifiy the tools by their functionalities showing some interesting results such as, for example, the fact that in spite of the apparent redundance between tools most of them perform slightly different tasks, suggesting that they all may be useful, or at least that redundance is more apparent than real.

References

1. Al-Shahrour F. Daz-Uriarte, R and J. Dopazo. Use of go terms to understand the biological significance of microarray differential gene expression data. In K. F. Johnson and S. M. Lin, editors, *Methods of microarray data analysis (CAMDA 2002)*, 2003.
2. S. Draghici, P. Khatri, R. P. Martins, G. C. Ostermeier, and S. A. Krawetz. Global functional profiling of gene expression. *Genomics*, 81(2):98–104, February 2003.
3. Steven Goodman. Commentary: The P-value, devalued. *Int. J. Epidemiol.*, 32(5):699–702, 2003.
4. J.L. Mosquera and A. Sánchez-Pla. Serbgo: Searching the best go tool. In *European Conference on Computational Biology*. ICBS, 2005.

W368–W371 *Nucleic Acids Research, 2008, Vol. 36, Web Server issue*
doi:10.1093/nar/gkn256

Published online 14 May 2008

SerbGO: searching for the best GO tool

J. L. Mosquera[1,2,*] and A. Sánchez-Pla[1,2]

[1]Statistics and Bioinformatics Research Group, Departament d'Estadística, Universitat de Barcelona. Av. Diagonal 645, 08028 Barcelona and [2]Statistics and Bioinformatics Unit, IRHUVH. Passeig Vall d'Hebron 119-129, 08035 Barcelona, Spain

Received February 3, 2008; Revised April 10, 2008; Accepted April 20, 2008

ABSTRACT

In recent years, the scientific community has provided many tools to assist with pathway analysis. Some of these programs can be used to manage functional annotation of gene products, others are oriented to exploring and analyzing data sets and many allow both possibilities. Potential users of these tools are faced with the necessity to decide which of the existing programs are the most appropriate for their needs. SerbGO is a user-friendly web tool created to facilitate this task. It can be used (i) to search for specific functionalities and determine which applications provide them and (ii) to compare several applications on the basis of different types of functionalities. Iterating and combining both functionalities can easily lead to selecting an appropriate tool. Data required by SerbGO is either the desired capabilities within a defined *Standard Functionalities* Set or the list of the tools to be compared. The analysis performed carries out a cross-classification that produces an easily readable output with the list of tools that implement the capabilities demanded or a table with the categorization of the GO tools that one wishes to compare. SerbGO is freely available and does not require a login. It can be accessed either directly at our server (http://estbioinfo.stat.ub.es/apli/serbgo) or at the GO Consortium website (http://www.geneontology.org/GO.tools.microarray.shtml#serbgo).

INTRODUCTION

Modern experimental technologies, such as DNA microarrays (1), have become both popular and affordable over the last decade, leading to a considerable increase in experiments and publicly available functional genomic data sets. These high-throughput methodologies pose different challenges: the experiment itself, the statistical analysis of the data and the obtention of biological knowledge from the data. For example, in gene-expression microarray studies, it is very common for the statistical analysis to yield long lists of genes and one of the main challenges is how to give these lists a biological interpretation (2). It might be reasonable to expect that this could be done relying on the information stored in the existing biological databases, which can help to relate the experimental results with previously existing biological knowledge.

A useful resource to achieve both the goal of interpretation and the need of automation is the Gene Ontology (GO) (3). The GO is a cooperative project, which was set in motion in the late 90s, developed and maintained by the GO Consortium. Briefly, it is an annotation database originated 'to provide a controlled vocabulary to describe gene and gene product attributes in any organism'. It consists of three independent ontologies: Biological Process (BP), Molecular Function (MF) and Cellular Component (CC). Each of them is represented as a directed acyclic graph (DAG) (4) with two kinds of relationships ('is-a' and 'part-of') and whose nodes are the GO terms arranged from the most specific ones at the bottom to the only one at the top which is the most general term. The gene products may be linked to one or more GO terms in these ontologies. Thus, when a given gene has been annotated to a GO term it is also linked to its related nodes.

In recent years, many tools have been developed to assist analysis of experimental results based on the GO. Some of these tools are intended to manage functional annotations while others are specific for analyzing gene lists and many allow both possibilities (5). The scientific community has rapidly moved from lacking the appropriate GO tools to having a wide range of applications with, seemingly, very similar capabilities. It seems reasonable to ask ourselves whether it is worthwhile to keep developing new variants of the same programs. We may have reached the point where most needs may be solved by an already existing tool and the problem is simply deciding between those tools available.

This article presents a web-based application called SerbGO (Searching for the best GO tool), intended to help users to select the tools which best suit their needs as well

*To whom correspondence should be addressed. Tel: +34 93 402 15 60; Fax: +34 93 411 17 33; Email: jlmosquera@ir.vhebron.net; jlmosquera@gmail.com as to easily compare the capabilities of various applications in the context of their experiments.

GO TOOLS AND THE STANDARD FUNCTIONALITIES SET

Due to the high heterogeneity among different types of tools it was decided to focus only on 'Tools for Gene Expression/Microarray Analysis' (http://www.geneontology.org/GO.tools.microarray.shtml).

To build SerbGO, a long list of applications available at the GO website (microarray tools) was reviewed from the existing literature. These tools use either the ontologies or the gene associations provided by the GO Consortium to facilitate the analysis of gene expression data.

The review yielded a substantial number of heterogeneous features, which were grouped into a potential set of functionalities. After several iterations, the features initially selected were converted into specific functionalities once redundancies were excluded. This process resulted in a set of features arranged in 205 standard functionalities.

The capabilities of the GO tools analyzed were classified *in situ* according to the *Standard Functionalities Set* and taking the following criteria into account:

(1) The functionality was available in the GO tool.
(2) The functionality was mentioned in the publication but it could not be validated.
(3) The functionality was not found in the paper or the application.

The list of applications which was finally included with their references is provided as Supplementary Material.

These tools use either the ontologies or the gene associations provided by the GO Consortium to facilitate the analysis of gene expression data. It must be noted that inclusion in the GO website does not imply approval by the GO Consortium and does not mean the tool has been tested or has been found to use information accurately. It can be said that this list 'is provided to promote an exchange of information between users and software developers'.

APPLICATION OUTLINE

Inputs

SerbGO is a web-based application designed to (i) facilitate researchers the task of determining which of the existing tools are appropriate for their needs and (ii) to enable a comparison between some of the available tools.

(1) The input needed to select those tools with the desired set of capabilities is a list of functionalities from the Standard Functionalities Set.
(2) The input needed to compare several tools is the list of programs to be compared.

Both actions can be performed interactively using the *Query Form* or the *Compare Tools* menu options (Figure 1).

Table 1. Number of standard functionalities per section

Section	No. of functionalities
Tools for	2
Type of experiment	7
Interface	7
Availability	4
Supported species	26
Data	40
Annotation	70
Statistical analysis	26
Output	23

Tools analyzed were classified according to a set of 205 standard functionalities arranged in nine sections.

Tool selection

The Query Form menu option at the top of the page allows the user to select different functionalities and to get the most appropriate tools to provide them. This form contains the *Standard Functionalities Set* arranged in nine sections (Table 1) and spread out over six pages.

To find the 'right tool' a user selects the desired functionalities by checking the appropriate fields at the specific sections (Figure 1A–C). Once the choices have been made for a page it is required to validate the query by clicking on the 'Next' button at the bottom of the page, which allows the user to move on the following one. The next page will show the new sections and the remaining tools will appear at the top-right corner. At the last selection page a 'Find' button will appear instead of 'Next' button. This new button allows users to move on to the outputs after validation.

Nonavailable features are shown as shaded colors. They can be activated by switching the corresponding radio button. In such cases, the user could have access to this option by switching on the previous radio button.

Queries are implemented with the logical operator AND. That is, the more capabilities are selected, the less tools will be available.

During the process of navigation over the pages, and at any time, it is possible to start a new query if the user clicks on the Query Form menu option at the top of the page.

Tool comparison

By checking any of the tools in the *Compare Tools* form, a list of their capabilities according to the *Standard Functionalities Set* can be obtained (Figure 1F).

Outputs

The output for the *Query Form* is a table with an alphabetically sorted list of the tools performing the functionalities demanded, the name of the developer and the name of the tool linked to its corresponding site (Figure 1D). The programs shown can be compared by clicking the Find button at the bottom of the results page (Figure 1E).

The output page for the Compare Tools form shows a table where rows contain the categorized functionalities and columns contain the GO tools names, which are linked to their respective sites (Figure 1E).

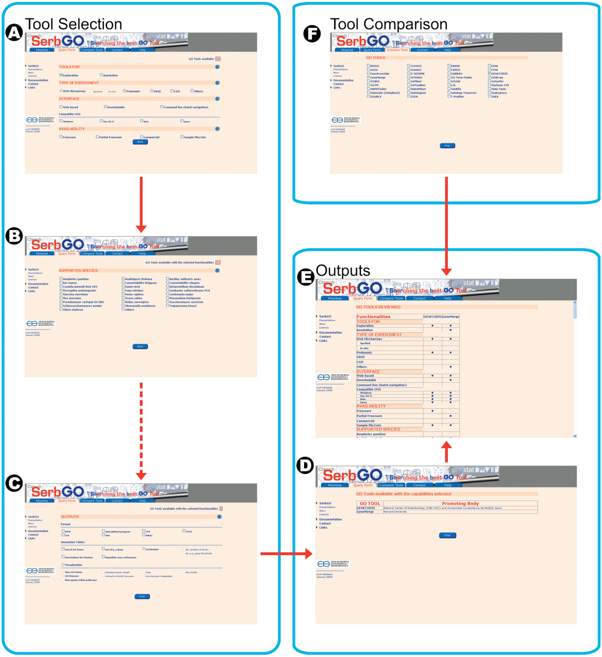


Figure 1. SerbGO workflow. (**A**) First page of the *Query Form* shows the *Standard functionalities* for the following sections: TOOL FOR, TYPE OF EXPERIMENT, INTERFACE and AVAILABILITY. (**B**) After the first validation a user selects the SUPPORTED SPECIES required and follows with the query until the last page. On the top-right corner is shown the number of tools available. (**C**) By clicking on the 'Find' button at the bottom of the page, the programs that fit the capabilities selected will be shown. (**D**) This screenshot shows the output for a list of tools and their developers. They can be compared if a user clicks on the 'Find' button. (**E**) A cross-tabulation for functionalities available in each tool is shown when the researcher requires a comparison of them. It can be attained either by comparing the output list of a *Query Form* or by selecting a set of tools at the *Compare Tools* form. (**F**) This page shows the entire collection of tools included in SerbGO. These programs can be compared by selecting the desired ones which query has to be validated on the button displayed at the bottom of the page.

Example

To illustrate the concept of how to determine which GO tools for gene-expression analysis provide the features required by a potential user the following example can be considered.

A potential SerbGO user has a list of *Drosophila melanogaster* genes. He/she would like to know which tools are available to (i) do a GO enrichment analysis (ii) that allow FlyBase Ids and (iii) correction for multiple testing for hypergeometric distribution tests. In such a situation, the user should click on the *Query Form* menu option and selects 'Exploration' at the TOOLS FOR section (Figure 1A). After that, move on the next page and selects 'Drosophila melanogaster' option (Figure 1B). When validation is made, there are 19 tools available. In DATA section, the user checks 'FlyBase ID' identifiers. He/she has to follow until the STATISTICAL ANALYSIS section, where will select 'Enrichment of GO Terms', 'Hypergeometric' test and 'Correction for Multiple Tests'. When the user gets the last query page, after clicking on the Find button the outputs are shown (Figure 1C). The researcher can see that there are two tools implementing the capabilities desired: GENECODIS and GeneMerge (Figure 1D). Now, if he/she wishes to compare the tools, it can be done by simply clicking on the new 'Find' button. This comparison will show a cross-tabulation of the capabilities available in GENECODIS and GeneMerge (Figure 1E).

IMPLEMENTATION AND AVAILABILITY

SerbGO is a web tool developed in PHP 4.3.3 on Windows using the ADOdb Database Abstraction Library for PHP and the Javascript language increased interactivity. It runs accurately on Mozilla Firefox, Internet Explorer and Konqueror browsers.

The information about tools and their functionalities has been stored in a database implemented in the open source relational database management system MySQL.

SerbGO is freely available under a Common Creative license and does not require a login. It can be accessed directly at our server (http://estbioinfo.stat.ub.es/apli/serbgo). The tool was submitted to the GO Consortium and is also available at their site (http://www.geneontology.org/GO.tools.microarray.shtml#serbgo).

BENCHMARK

SerbGO has been running since June 2006. During the testing period, most of the tools available at the GO Consortium website were included in the beta version. This version was used by several people outside the authors. SerbGO was also tested by the developers of some of the tools such as FatiGO or GARBAN who suggested some improvements that were incorporated into the testing version and validated at the first stable version.

DISCUSSION

Whether because of a lack of information about what GO tools do or because of the large number of applications available, it has long seemed reasonable for researchers to implement their own tools to 'provide' biological meaning for their experiments. This has resulted in many, and often very similar programs, which has surfaced the need for an application such as SerbGO that can be used to explore and differentiate amongst the ever-growing set of GO tools.

Thanks to the Standard Functionalities Set, a GO tool can be easily classified to determine which capabilities it implements. This greatly facilitates the task of choosing a tool that adapts to the specific interest of a user. SerbGO is intended to be used by experimental biologists without any previous training in bioinformatics. However, it should be taken into account that the best search approach is to start by checking few capabilities and in subsequent iterations gradually increase the features of interest until a satisfying list of tools is obtained. In other words, the main idea is not to check all the capabilities required at once, since this may result in a null output.

SerbGO is the only web tool to proceed in such a way and after 2 years we have observed that it is highly flexible to obtain an application or a set of applications that allow the researcher to attain their goals. In order to keep SerbGO useful, it is updated periodically (twice a year at least) and accurately. Users, especially GO tool developers, are welcome to help us implement improvements to SerbGO.

SUPPLEMENTARY DATA

Supplementary Data are available at NAR Online.

ACKNOWLEDGEMENTS

The authors would like to thank all members of the Statistics and Bioinformatics Research Group, Department of Statistics, University of Barcelona, for their useful suggestions and comments. We are also grateful to Dr Fátima Nuńez from the IRHUVH for her suggestions. Funding to pay the Open Access publication charges for this article was provided by the project no. 303483 FBG-University of Barcelona.

Conflict of interest statement. None declared.

REFERENCES

1. Simon,R.M., Korn,E.L., McShane,L.M., Radmacher,M.D., Wright,G.W. and Zhao,Y. (2004) *Design and Analysis of DNA Microarray Investigations*. Springer-Verlag, New York.
2. Sánchez-Pla,A. and Mosquera,J.L. (2008) The quest for biological significance. In Bonilla,L.L., Moscoso,M., Platero,G., Vega,J.M. (ed.), *Progress in Industrial Mathematics at ECMI 2006*. Series Mathematics in Industry. Subseries The European Consortium for Mathematics in Industry. Springer, Heidelberg, vol. 12, pp. 566–570.
3. Asburner,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. *et al.* (2000) Gene Ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat. Genet.*, **25**, 25–29.
4. Diestel,R. (2000) *Graph Theory*. Springer-Verlag, New York.
5. Khatri,P. and Drăghici,S. (2005) Ontological analysis of gene expression data: current tools, limitations, and problems. *Bioinformatics*, **18**, 3587–3595.

`sims`: an `R` package for Computing Semantic Similarities

Jose Luis Mosquera and Alex Sánchez

August 31, 2014

Contents

1 Introduction

An ontology is a way for annotating concepts of a certain domain. It allows the comparison between entities through their associated concepts, and which otherwise would not be comparable. The structure of the vocabulary of an ontology is arranged as a rooted directed acyclic graph (DAG). That is, an ontology is a hierarchy with a single "highest" term called the *root*. All other descendant terms are connected by either one or a several directed links (i.e. the links point upwards) to the root, an these links are acyclic (i.e. cycles are not allowed in the graph).

One of the most successful ontologies for annotating biological vocabularies is the Gene Ontology (GO). It is an annotation resource created and maintained by a public consortium, http://geneontology.org/page/go-consortium-contributors-list [18]. The main goal of the consortium is *citing their mission,* *to produce a controlled vocabulary that can be applied to all organisms even as knowledge of gene and protein roles in cells is accumulating and changing.* It is organized covering three domains: *Cellular Component* (*CC*), *Biological Process* (*BP*), and *Molecular Function* (*MF*). Each ontology domain consists of a high number of terms or categories hierarchically related from least (top) to most (bottom) specialized characteristics.The GO has two types of relationship (i.e. links)between GO terms: the *is-a* and the *part-of*.

Usually, an ontology is used for the interpretation of sets of objects mapped to this ontology. For example, the GO allows annotating genes and their products. Most genes are annotated in one or more GO terms. Annotations are made as specific as possible. As a consequence a gene is associated not only with its annotations but also with all the less specific terms associated with them. Furthermore, a given gene product may represent one or more molecular functions, be used in one or more biological processes and appear in one or more cellular components.

Many applications using ontologies require to determine the relationship between pairs of terms [11, 12]. An appropriate measure of such relationship is the semantic similarity between the terms. Generally speaking, a semantic similarity between two terms is as a function of distance between the terms in the graph corresponding to the underlying ontology [3]. There are different methods a approaches [4], but mainly they are classified into (1) methods based on node-based approaches, (2) methods based on edge-based approaches, and (3) methods based on hybrid-based approaches.

`sims` package provides functions for dealing with arbitrary ontologies, computing semantic similarities between their and comparing lists of objects annotated in these ontologies, particularly focused on the GO.

The present present document is just an introduction to the use of `sims` package.

To start with `sims` package, write the following code

```
library("sims")
help("sims")
```

Functions available in the package are

```
ls("package:sims")

##  [1] "ancestors"         "commonAncestors"
##  [3] "cosSim"            "depth"
##  [5] "distRada"          "getA"
##  [7] "getGk"             "getGr"
##  [9] "GOANCESTORS"       "goOOC"
## [11] "GOPARENTS"         "gosims"
## [13] "gosimsAvsB"        "gosimsProfiles"
## [15] "ICA"               "inverseIminusG"
## [17] "is.OOC"            "LCAs"
## [19] "mapEG2GO"         "mappingMatrix"
## [21] "Nt"               "pdHap"
## [23] "pdHax"            "pdHm"
## [25] "pdHx"             "plotGODAG"
## [27] "plotHistSims"     "pseudoDists"
## [29] "refinementMatrix" "resnikSummary"
## [31] "simFaith"         "simJC"
## [33] "simLin"           "simNunivers"
## [35] "simPsec"          "simRada"
## [37] "simRel"           "simRes"
## [39] "simRes.eb"        "simsBetweenGOIDs"
## [41] "sims.eb"          "simsMat"
## [43] "sims.nb"          "summaryMICA"
## [45] "summaryPaths"     "summarySims"
## [47] "summarySimsAvsB"  "termPairs"
## [49] "toMat"            "toOOC"
## [51] "toPairs"
```

2 Semantic Similarities Between Terms of an Arbitrary Ontology Mapped by a List of Objects

To illustrate the usage of basic structures and the computation of semantic similarities between terms of an arbitrary ontology with `sims` package, we are going to make use of an example proposed by Joslyn *et al* [6]. It consists of a 10 object identifiers mapping to terms of an ontology with 12 concepts. Figure 1 shows a representation of the example considered.

In order to deal with the structure we make use of a concept called *Object-Ontology Complex* (*OOC*) introduced by Carey [1], that we will see in next section 2.1. But, previously we need to “translate” the graph structure described above in terms of matrices.

For the inpatient user, load the following dataset into memory in order to compute semantic similarities and goes to subsection 2.2

```
data(joslyn)
help("joslyn")
```

Otherwise, next coding lines provides the process for building the matrix forms associated with the each component of the structure presented above

```
## 1. Vocabulary of the ontology
vocabulary <- c("R", "B", "C", "K", "F", "G", "I", "E", "J", "H",
"A", "D")
```

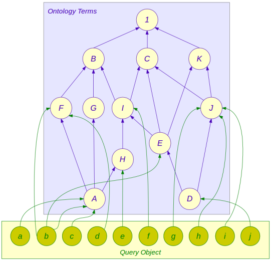


Figure 1: Representation of an ontology with 12 terms and 10 object identifiers annotated in the ontology

```
## 2. Links between terms (structure of the ontology)
origin <- c("B", "C", "K", "F", "G", "I", "I", "E", "J", "E", "J",
"A", "A", "E", "H", "D", "D", "A")
terminus <- c("R", "R", "R", "B", "B", "B", "C", "C", "C", "K",
"K", "F", "G", "I", "I", "E", "J", "H")
links <- data.frame(origin, terminus)
mat.g <- toMat(df = links, rnames = vocabulary,
cnames = vocabulary)

print(mat.g)

##   R B C K F G I E J H A D
## R 0 0 0 0 0 0 0 0 0 0 0 0
## B 1 0 0 0 0 0 0 0 0 0 0 0
## C 1 0 0 0 0 0 0 0 0 0 0 0
## K 1 0 0 0 0 0 0 0 0 0 0 0
## F 0 1 0 0 0 0 0 0 0 0 0 0
## G 0 1 0 0 0 0 0 0 0 0 0 0
## I 0 1 1 0 0 0 0 0 0 0 0 0
## E 0 0 1 1 0 0 1 0 0 0 0 0
## J 0 0 1 1 0 0 0 0 0 0 0 0
## H 0 0 0 0 0 0 1 0 0 0 0 0
## A 0 0 0 0 1 1 0 0 0 1 0 0
## D 0 0 0 0 0 0 0 1 1 0 0 0

## 3. Objects identifiers that are annotates in the ontology
object.ids <- letters[1:10]

## 4. Mapping from objects to terms (annotation of objects)
object <- c("b", "d", "f", "b", "g", "h", "i", "e", "a", "b", "c",
"j")
term <- c("F", "F", "I", "E", "J", "J", "J", "H", "A", "A", "A",
"D")
map <- data.frame(object, term)
mat.m <- toMat(df = map, rnames = object.ids, cnames = vocabulary)

print(mat.m)

##   R B C K F G I E J H A D
## a 0 0 0 0 0 0 0 0 0 0 1 0
## b 0 0 0 0 1 0 0 1 0 0 1 0
## c 0 0 0 0 0 0 0 0 0 0 1 0
## d 0 0 0 0 1 0 0 0 0 0 0 0
## e 0 0 0 0 0 0 0 0 0 1 0 0
## f 0 0 0 0 0 0 1 0 0 0 0 0
## g 0 0 0 0 0 0 0 0 1 0 0 0
## h 0 0 0 0 0 0 0 0 1 0 0 0
## i 0 0 0 0 0 0 0 0 1 0 0 0
## j 0 0 0 0 0 0 0 0 0 0 0 1
```

2.1 Object-Ontology Complex (OOC) Container

An OOC is *a formalism for working with ontologies for statistical purposes*. It combines the four elements described in previous section 2. That is, (1) the terms of the ontology, (2) the structure of the directed acyclic graph (DAG), (3) the list of objects annotated in the ontology, and (4) how the objects map to the terms.

`sims` package has a class `OOC`, that is used as a general container for Object-Ontology Complexes (OOC).

```
help("OOC")
```

The function `toOOC` facilitates the construction of an object of class `OOC`. This object is merely used as a container of the elements of the OOC. It has four slots `T` (the list of terms or vocabulary of the ontology), `G` (the matrix accessibility matrix or the matrix of 1-step refinement associated with DAF structure of the ontology), `O` (the list of object identifiers), and `M` (the mapping matrix between objects and terms).

```
joslyn.OOC <- toOOC(T = vocabulary, G = mat.g, O = object.ids,
                    M = mat.m)
print(joslyn.OOC)

## An object of class "OOC"
## Slot "T":
##  [1] "R" "B" "C" "K" "F" "G" "I" "E" "J" "H" "A" "D"
##
## Slot "G":
##   R B C K F G I E J H A D
## R 0 0 0 0 0 0 0 0 0 0 0 0
## B 1 0 0 0 0 0 0 0 0 0 0 0
## C 1 0 0 0 0 0 0 0 0 0 0 0
## K 1 0 0 0 0 0 0 0 0 0 0 0
## F 0 1 0 0 0 0 0 0 0 0 0 0
## G 0 1 0 0 0 0 0 0 0 0 0 0
## I 0 1 1 0 0 0 0 0 0 0 0 0
## E 0 0 1 1 0 0 1 0 0 0 0 0
## J 0 0 1 1 0 0 0 0 0 0 0 0
## H 0 0 0 0 0 0 1 0 0 0 0 0
## A 0 0 0 0 1 1 0 0 0 1 0 0
## D 0 0 0 0 0 0 0 1 1 0 0 0
##
## Slot "O":
##  [1] "a" "b" "c" "d" "e" "f" "g" "h" "i" "j"
##
## Slot "M":
##   R B C K F G I E J H A D
## a 0 0 0 0 0 0 0 0 0 0 1 0
## b 0 0 0 0 1 0 0 1 0 0 1 0
## c 0 0 0 0 0 0 0 0 0 0 1 0
## d 0 0 0 0 1 0 0 0 0 0 0 0
## e 0 0 0 0 0 0 0 0 0 1 0 0
## f 0 0 0 0 0 0 1 0 0 0 0 0
## g 0 0 0 0 0 0 0 0 1 0 0 0
## h 0 0 0 0 0 0 0 0 1 0 0 0
## i 0 0 0 0 0 0 0 0 1 0 0 0
## j 0 0 0 0 0 0 0 0 0 0 0 1
```

2.2 Comptutation of Semantic similarities

In sims package there are implemented a total of fourteen measures from different approaches. The following subsections describe main functions to compute semantic similarities between all the pairs of terms of the induced graph (from the ontology) by a list of object identifiers.

2.2.1 Methods of Node-Based Approach

There are implemented seven semantic similarity measures proposed by Resnik [13], Lin [7], Schlicker *et al.* [15], Jiang and Conrath [5], Mazandu and Mulder [9], Pirró and Seco [11], and Pirró and Euzenat [10]. All the methods are based on the concept of *Information Content* (*IC*) proposed by Resnik [13], and the shared information between the two terms being measured is proportional to the IC of the Most Informative Common Ancestor (MICA) in the rooted DAG.

Semantic similarities measures of node-based approach are computed by calling the wrapper function `sims`.

```
help("sims")
```

Three arguments are required by this function: a `list` with the ancestors of each selected term (`at`), a `numeric` vector with the IC of each term (`ic`), and the `method` required (see possibilities in the help).

To obtain the list of ancestors we need to build the accessibility matrix associated with the DAG structure by performing the following computation

```
## Accessibility matrix
inv.IminusG <- inverseIminusG(joslyn.OOC)
A.mat <- getA(inv.IminusG)
print(A.mat)

##       R     B     C     K     F     G     I     E     J
## R FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## B  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## C  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## K  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## F  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## G  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## I  TRUE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE
## E  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE FALSE FALSE
## J  TRUE FALSE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE
## H  TRUE  TRUE  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE
## A  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE FALSE FALSE
## D  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE  TRUE  TRUE
##       H     A     D
## R FALSE FALSE FALSE
## B FALSE FALSE FALSE
## C FALSE FALSE FALSE
## K FALSE FALSE FALSE
## F FALSE FALSE FALSE
## G FALSE FALSE FALSE
## I FALSE FALSE FALSE
## E FALSE FALSE FALSE
## J FALSE FALSE FALSE
## H FALSE FALSE FALSE
## A  TRUE FALSE FALSE
## D FALSE FALSE FALSE
```

```
## Ancestors
at <- ancestors(A.mat)
print(at)
```

```
## $R
## [1] "R"
##
## $B
## [1] "R" "B"
##
## $C
## [1] "R" "C"
##
## $K
## [1] "R" "K"
##
## $F
## [1] "R" "B" "F"
##
## $G
## [1] "R" "B" "G"
##
## $I
## [1] "R" "B" "C" "I"
##
## $E
## [1] "R" "B" "C" "K" "I" "E"
##
## $J
## [1] "R" "C" "K" "J"
##
## $H
## [1] "R" "B" "C" "I" "H"
##
## $A
## [1] "R" "B" "C" "F" "G" "I" "H" "A"
##
## $D
## [1] "R" "B" "C" "K" "I" "E" "J" "D"
```

Function `resnikSummary` builds a `data.frame` providing the number of times that each term or any of its refinements appears in the OOC (i.e. $n(t_i)$), the probability of finding the term (i.e. $p(t_i)$), and the Information Content of the term (i.e. $IC(t_i)$). Thus, we can calculate the IC's of each term very easily by performing

```
resnik.sum <- resnikSummary(x = joslyn.OOC)
print(resnik.sum)

##   nt      pt     ic
## R 34 1.00000 0.0000
## B 15 0.44118 0.8183
## C 13 0.38235 0.9614
## K  6 0.17647 1.7346
## F  5 0.14706 1.9169
## G  3 0.08824 2.4277
## I  7 0.20588 1.5805
## E  2 0.05882 2.8332
## J  4 0.11765 2.1401
## H  4 0.11765 2.1401
## A  3 0.08824 2.4277
## D  1 0.02941 3.5264

ic <- resnik.sum[, "ic"]
```

Finally, to compute the semantic similarity we just only indicate the method required

```
## Computation of semantic similarity of Resnik
## (node-based approach)
sims.Res <- sims.nb(at, ic, method = "Res")
head(sims.Res)

##     Resnik
## B-R      0
## C-R      0
## K-R      0
## F-R      0
## G-R      0
## I-R      0

## Computation of all semantic similarities of
## edge-based approach
sims.all <- sims.nb(at, ic, method = "all")
head(sims.all)

##     Resnik Lin Rel     JC Nunivers    Psec Faith
## B-R      0   0   0 0.5500        0 -0.8183     0
## C-R      0   0   0 0.5098        0 -0.9614     0
## K-R      0   0   0 0.3657        0 -1.7346     0
## F-R      0   0   0 0.3428        0 -1.9169     0
## G-R      0   0   0 0.2917        0 -2.4277     0
## I-R      0   0   0 0.3875        0 -1.5805     0
```

The following function provides a summary of the measures

```
summarySims(sims.all)

##             n NAs     Min Num.Min    Max Num.Max    Mean
## Resnik   66   0  0.0000      24 2.8332       1  0.7998
## Lin      66   0  0.0000      24 1.0000       1  0.3723
## Rel      66   0  0.0000      24 0.9118       1  0.2661
## JC       66   0  0.1796       1 1.0000       1  0.3662
## Nunivers 66   0  0.0000      24 0.8034       1  0.2268
## Psec     66   0 -4.5678       1 2.4277       1 -1.3518
## Faith    66   0  0.0000      24 1.0000       1  0.2804
##          Std.Dev  Median
## Resnik    0.7399  0.8183
## Lin       0.3210  0.4146
## Rel       0.2609  0.2393
## JC        0.1582  0.3271
## Nunivers  0.2098  0.2321
## Psec      1.6449 -1.4746
## Faith     0.2707  0.2616
```

2.2.2 Methods of Edge-Based Approach

With regard to the edge-based approach there are implemented two semantic similarity measures proposed by Resnik [13] and Rada *et al.* [12]. But also, it is a distance measure proposed by Rada [12], and four pseudo-distances proposed by Joslyn *et al* [6].

Semantic similarities measures of edge-based approach are computed by calling the wrapper function `sims.eb`.

```
help("sims.eb")
```

This function depends on four arguments: the `OOC` object (`x`), the name of the root term of the ontology (`root`), the `list` of the ancestors of each selected term (`at`), and the `method` required (see possibilities in the help).

```
## Computation of semantic similarity of Resnik
## (edge-based approach)
Resnik.eb <- sims.eb(x = joslyn.OOC, root = "R", at,
                     method = "Rada")
head(Resnik.eb)

##       Rada
## B-R 0.5000
## C-R 0.5000
## K-R 0.5000
## F-R 0.3333
## G-R 0.3333
## I-R 0.3333

## Computation of all semantic similarities of
## edge-based approach
sims.eb.all <- sims.eb(x = joslyn.OOC, root = "R", at,
                       method = "all")
head(sims.eb.all)

##       Rada Resnik.eb
## B-R 0.5000         7
## C-R 0.5000         7
## K-R 0.5000         7
## F-R 0.3333         6
## G-R 0.3333         6
## I-R 0.3333         6

## Summary of semantic similarities
summarySims(sims.eb.all)

##             n NAs Min Num.Min Max Num.Max   Mean Std.Dev
## Rada       66   0 0.2       8 0.5      18 0.3399  0.1084
## Resnik.eb 66   0 4.0       8 7.0      18 5.7576  0.9932
##           Median
## Rada      0.3333
## Resnik.eb 6.0000
```

Distance measure of Rada can be computed by calling the function `distRada`.

```
help("distRada")
```

The function requires a `list` of `numeric` vectors with the lengths (in terms of depth) of the number of paths between each pair of terms (`sum.paths`), and the `list` of the ancestors of each selected term (`at`). To obtain the first argument we make use of the function `summaryPaths`

```
sum.paths <- summaryPaths(x = joslyn.OOC, root = "R", len = TRUE)
head(sum.paths, 10)
```

```
##     [,1] [,2] [,3] [,4]
## R-R    0    0    0    0
## B-R    1    0    0    0
## C-R    1    0    0    0
## K-R    1    0    0    0
## F-R    0    2    0    0
## G-R    0    2    0    0
## I-R    0    2    0    0
## E-R    0    2    3    0
## J-R    0    2    0    0
## H-R    0    0    3    0
```

Then, distance is calculated by

```
Rada <- distRada(sum.paths, at)
head(Rada)

##     sp.Rada
## B-R       1
## C-R       1
## K-R       1
## F-R       2
## G-R       2
## I-R       2

summarySims(Rada)

##          n NAs Min Num.Min Max Num.Max  Mean Std.Dev Median
## sp.Rada 66   0   1      18   4       8 2.242  0.9932      2
```

Pseudo-distances implemented in `sims` package can be computed by calling the function `pseudoDists`.

```
help("pseudoDists")
```

This function needs to be fed with the `OOC` object (`x`), the name of the root term of the ontology (`root`), and the `method` required (see possibilities in the help).

```
## Computation of the pseudo-distance of
## the minimum chain length
pd.hm <- pseudoDists(x = joslyn.OOC, root = "R", method = "hm")
head(pd.hm)

##     h.m
## B-R   1
## C-R   1
## K-R   1
## F-R   2
## G-R   2
## I-R   2

## Computation of all pseudo-distance
pd.all <- pseudoDists(x = joslyn.OOC, root = "R", method = "all")
head(pd.all)

##     h.m h.x h.ax h.ap
## B-R   1   1    1    1
## C-R   1   1    1    1
## K-R   1   1    1    1
## F-R   2   2    2    2
## G-R   2   2    2    2
## I-R   2   2    2    2

## Summary of pseudo-distances
summarySims(pd.all)

##       n NAs Min Num.Min Max Num.Max  Mean Std.Dev Median
## h.m  66  30   1      18 3.0       5 1.639  0.7232   1.50
## h.x  66  30   1      17 4.0       2 1.806  0.9202   2.00
## h.ax 66  30   1      17 3.5       2 1.722  0.8057   1.75
## h.ap 66  30   1      17 3.5       2 1.722  0.8057   1.75
```

3 Semantic Similarities Associated with the GO

The package can manage any ontology, but it is especially focused on the Gene Ontology. In this regard, there are some functions that are particularly adapted for allow building the refinements matrix (i.e. the accessibility matrix) and the mapping matrix (i.e the matrix that maps from Entrez Gene IDs to GO IDs), performing comparisons between lists of semantic similarities, and yield different types of plots (e.g. histograms, diagram bars and DAG's of the induced graphs). Moreover, `sims` package can manage Entrez Gene IDs and GO IDs from any `R` organism package.

In order to explore and compare semantic similarities the package takes advantage of two experimental datasets from two prostate cancer experiments [19] and [16], provided by the `R` package `goProfiles` [14]. Thus, first of all, a dataset with several lists of genes, from two different studies, selected as being differentially expressed in prostate cancer is loaded into memory

```
data(prostateIds)
help("prostateIds")

## No documentation for 'prostateIds' in specified packages and libraries:
## you could try '??prostateIds'
```

Then, two subsets of Entrez Gene ID's are selected from two different lists of genes respectively.

```
## Entrez Gene ID's from Welsh et al. study
eg.we <- welsh01EntrezIDs[1:10]

## Entrez Gene ID's from Singh et al. study
eg.sg <- singh01EntrezIDs[1:10]
```

And finally, provide the name of human `R` organism package

```
pckg <- "org.Hs.eg.db"
```

3.1 Semantic similarities between GO IDs ancestors of terms that have been mapped by Entrez Genes

Function `gosims` allows to compute semantic similarities between all the pairs of GO ID ancestors of terms that annotate the selected Entre Gene ID's.

```
help("gosims")
```

The function requires the list of genes (`eg`), the ontology domain (`ontology`), the name of the organism package (`pckg`), the type of approach (`type`), and the measure used (`method`). In this example are considered all the measures from node-based approach to compute semantic similarities between GO ID's of Molecular Function (`MF`) associated with the subset of genes selected from the Welsh *et al.* study

```
## All semantic similarities of node-based approach
all.nb <- gosims(eg = eg.we, ontology = "MF", pckg = pckg,
                 type = "nb", method = "all")

## Loading required package:  org.Hs.eg.db

summarySims(all.nb)

##              n NAs     Min Num.Min    Max Num.Max    Mean
## Resnik    6903   0 0.00000    4469 4.7005      10 0.45654
## Lin       6903   1 0.00000    4468 1.0000      42 0.11142
## Rel      6903   1  0.00000    4468 0.9909      10  0.08649
## JC       6903   0  0.09614     452 1.0000      43  0.14564
## Nunivers 6903   0  0.00000    4469 1.0000      10  0.09713
## Psec     6903   0 -9.40096     452 4.7005      10 -6.45551
## Faith    6903   1  0.00000    4468 1.0000      42  0.07772
##          Std.Dev  Median
## Resnik   0.84936  0.0000
## Lin      0.21059  0.0000
## Rel      0.19429  0.0000
## JC       0.09717  0.1202
## Nunivers 0.18070  0.0000
## Psec     2.64582 -7.3215
## Faith    0.17062  0.0000
```

3.2 Semantic similarities profiles

The following functions are though for performing comparisons between two semantic similarity profiles generated according two list of genes.

The reason for comparin two lists of semantic similarities may be to understand functional gene similarities. In order to perform this type of comparison, existing packages (e.gs `GOSim` [2] and `GOSemSim` [20]) propose different approaches based on similarities that yield judgments of orientation, but not magnitudes. `sims` package considers alternative strategies that rely on a more statistical approach. Some functions allow building summaries with magnitude measures and plots for highlighting differences between profiles. The following subsections illustrate the main ideas with an example that considers the two lists of genes subsetted from the studies of Welsh *et al.* and Singh *et al.*

3.2.1 Computation of the semantic similarity profiles

To compute the semantic similarity profiles associated with each list of Entrez Gene ID's we use the function `gosimsAvsB`. It looks for the induced graph given by two lists of Entrez Gene ID's annotated in the ontology domain, and then calculates the semantic similarities between all the pairs of GO ID ancestors associated with the GO ID's that are annotating each list of genes. Figure 2 shows the schematically the idea of this step

```
## Semantic similarity profiles computed with Resnik's measure
## from node-based approach
WEvsSG.nb <- gosimsAvsB(eg1 = eg.we, eg2 = eg.sg, ontology = "MF",
                        pckg = pckg, type = "nb", method = "Res")
```

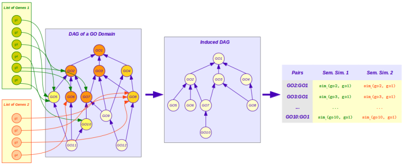


Figure 2: Schema for computing the two semantic similarities profiles associated with the two lists of genes respectively.

3.2.2 Comparison between the semantic similarity profiles

Statistical analysis is performed with the function `summarySimsAvsB`. It yields a summary that consists of (1) an statistic descriptive for each profile of semantic similarity measures, (2) a Mantel's Test [8] for examining the association between the distance matrices (i.e. the similarity matrices), and (3) a Cosine Similarity [17] for determining the similarity between the two semantic similarity profiles.

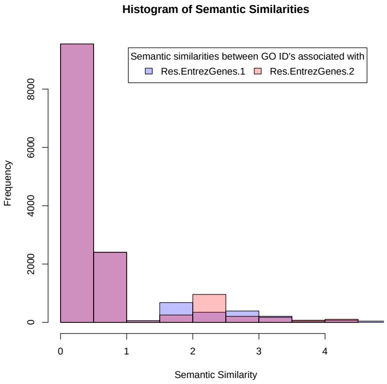
```
summarySimsAvsB(WEvsSG.nb)

## $Summary
##                         n NAs Min Num.Min   Max Num.Max
## Res.EntrezGenes.1 13861  59   0    9550 4.700      44
## Res.EntrezGenes.2 13861  73   0    9550 4.431     107
##                     Mean Std.Dev Median
## Res.EntrezGenes.1 0.4267  0.8552      0
## Res.EntrezGenes.2 0.4294  0.8713      0
##
## $Mantel
##   Mantel.r PValue
## 1   0.9698  0.001
##
## $Similarity
## [1] 0.9757
```

3.2.3 Plots for the semantic similarity profiles

In `sims` package there are three types of plots implemented. They support the statistical summary provided by the function `summaryAvsB`.

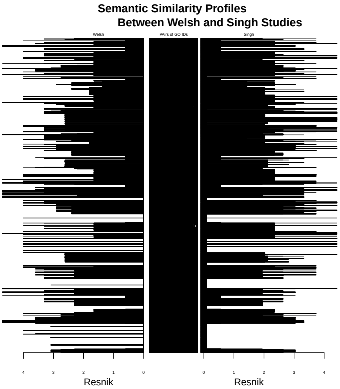
First plot is an histogram of the semantic similarity profiles. It shows both "curves" in the same plot.

```
plotHistSims(x = WEvsSG.nb, freq = TRUE,
             main = "Histogram of Semantic Similarities",
             xlab = "Semantic Similarity")
```



Second image plots a vertical bar diagram, whose bars are associated with the semantic similarities between each pair of terms. Bars on the left side are the bars corresponding to the first list of genes and bars on the right side are the bars corresponding to the second list of genes.

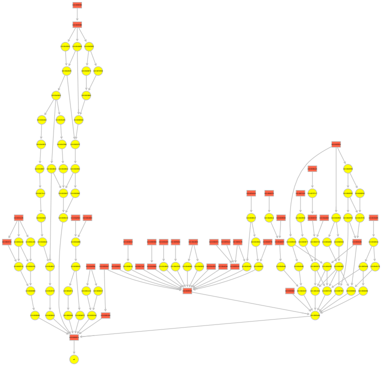
```
gosimsProfiles(x = WEvsSG.nb,
               col = c("tomato", "blue"), cex = 0.4,
               top.labels = c("Welsh", "PAirs of GO IDs", "Singh"),
               main = "Semantic Similarity Profiles
                       Between Welsh and Singh Studies",
                    xlab = "Resnik")
```



```
## [1] 5.1 4.1 4.1 2.1
```

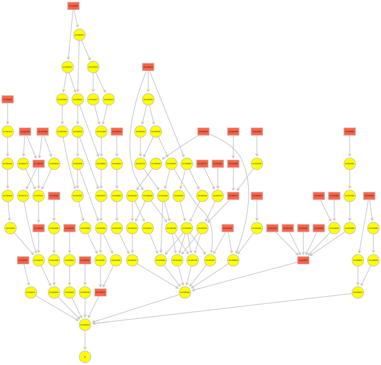
Function `plotDAG` plots the induced subgraph from the GO domain associated with one or two lists of Entrez Gene Identifiers. The subgraph shows two types of shapes for each node. Circles are GO ID's not mapped directly by the genes and rectangles are GO ID's that are mapped directly by the genes. The color of nodes indicate the type of relation with the Entrez Gene IDs. That is, when argument `eg2` is `NULL`, there are two possibilities: nodes mapped directly are shown in `red` color and their ancestors are shown in `yellow` color. But, if argument `eg2` is not `NULL`, then there are six different colors. Nodes mapped directly from the first list of Entrez Gene IDs are shown in `red` color and their ancestors are shown in `yellow` color. Nodes mapped directly from the second list of Entrez Gene IDs are shown in `lightblue` color and their ancestors are shown in `blue` color. Nodes mapped directly from both lists of Entrez Gene IDs are shown in `magenta` color and their ancestors are shown in `violet` color.

```r
## Induced subgraph associated with the list of genes from
## Welsh study
plotGODAG(eg1 = eg.we, eg2 = NULL, pckg = pckg, ontology = "MF")
```

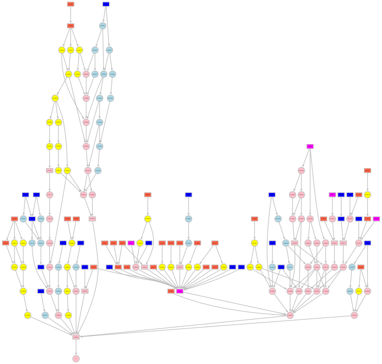


```
## [1] "A graph with 118 nodes."
```

```r
## Induced subgraph associated with the list of genes from
## Singh study
plotGODAG(eg1 = eg.sg, eg2 = NULL, pckg = pckg, ontology = "MF")
```



```
## [1] "A graph with 105 nodes."

## Induced subgraph associated with both lists of genes
plotGODAG(eg1 = eg.we, eg2 = eg.sg, pckg = pckg, ontology = "MF")
```



```
## [1] "A graph with 167 nodes."
```

References

[1] Vincent J. Carey. Ontology concepts and tools for statistical genomics. *Journal of Multivariate Analysis*, 90:213–228, 2003.

[2] Holger Fröhlich, Nora Speer, Annemarie Poustka, and Tim Beißbarth. GOSim - an R-package for computation of information theoretic GO similarities between terms and gene products. *BMC Bioinformatics*, 8(166), 2007.

[3] Mingxin Gan, Xue Dou, and Rui Jiang. From ontology to semantic similarity: Calculation of ontology-based semantic similarity. *The Scientific World Journal*, (793091):1–11, 2013.

[4] Sébastien Harispe, Sylvie Ranwez, Stefan Janaqi, and Jacky Montmain. Semantic measures for the comparison of units of language, concepts or entities from text and knowledge base analysis. *arXiv*, (1310.1285), 2013.

[5] Jay J. Jiang and David W. Conrath. Semantic similarity based on corpus statistics and lexical taxonomy. In *Proceedings of International Conference Research on Computational Linguistics (ROCLING X)*, pages 19–33, 1997. Tapei, Taiwan.

[6] Cliff A. Joslyn, Susan M. Mniszewski, Andy W. Fulmer, and Gary G. Heaton. The gene ontology categorizer. *Bioinformatics*, 20(s1):169–77, 2004.

[7] Dekang Lin. An information-theoretic definition of similarity. In *Proceedings of the Fifteenth International Conference on Machine Learning*, pages 296–304. Morgan Kaufmann Publishers, 1998.

[8] Nathan Mantel. The detection of disease clustering and a generalized regression approach. *Cancer Research*, 27(2 Part 1):209–220, 1967.

[9] Gaston K. Mazandu and Nicola J. Mulder. Information content-based gene ontology semantic similarity approaches: Toward a unified framework theory. *BioMed Research International*, (292063):1–11, 2013.

[10] Giuseppe Pirró and Jérôme Euzenat. A feature and information theoretic framework for semantic similarity and relatedness. In Peter F. Patel-Schneider, Yue Pan, Pascal Hitzler, Peter Mika, Lei Zhang, Jeff Z. Pan, Ian Horrocks, and Birte Glimm, editors, *The Semantic Web - ISWC 2010*, volume 6496 of *Lecture Notes in Computer Science*, pages 615–630. Springer Berlin Heidelberg, 2010.

[11] Giuseppe Pirró and Nuno Seco. Design, implementation and evaluation of a new semantic similarity metric combining features and intrinsic information content. In Robert Meersman and Zahir Tari, editors, *On the Move to Meaningful Internet Systems: OTM 2008*, volume 5332 of *Lecture Notes in Computer Science*, pages 1271–1288. Springer Berlin Heidelberg, 2008.

[12] Roy Rada, Hafedh Mili, Ellen Bicknell, and Maria Blettner. Development and application of a metric on semantic nets. *Systems, Man and Cybernetics, IEEE Transactions on*, 19(1):17–30, 1989.

[13] Phillip Resnik. Using information content to evaluate semantic similarity in a taxonomy. pages 448–453. Int. Joint Conf. on Artificial Intelligence, Kaufmann, Morgan, 1995.

[14] Alex Sánchez, Jordi Ocaña, and Miquel Salicrú. *goProfiles: goProfiles: an R package for the statistical analysis of functional profiles*, 2010. R package version 1.24.0.

[15] Andreas Schlicker, Francisco S. Domingues, Jörg Rahnenführer, and Thomas Lengauer. A new measure for functional similarity of gene products based on Gene Ontology. *BMC Bioinformatics*, 7:302, 2006.

[16] Dinesh Singh, Philip K. Febbo, Kenneth Ross, Donald G. Jackson, Judith Manola, Christine Ladd, Pablo Tamayo, Andrew A. Renshaw, Anthony V. DAmico, Jerome P. Richie, Eric S. Lander, Massimo Loda, Philip W. Kantoff, Todd R. Golub, and William R. Sellers. Gene expression correlates of clinical prostate cancer behavior. *Cancer Cell*, 1(2):203–209, 2002.

[17] Pang-Ning Tan, Michael Steinbach, and Vipin Kumar. *Introduction to Data Mining*. Addison-Wesley Longman Publishing Co., Inc., firt edition edition, 2005.

[18] The Gene Ontology Consortium. The gene ontology (go) database and informatics resource. *Nucleic Acids Research*, 32(Suppl. 1):D258–D261, 2004.

[19] John B. Welsh, Lisa M. Sapinoso, Andrew I. Su, Suzanne G. Kern, Jessica Wang-Rodriguez, Christopher A. Moskaluk, Henry F. Frierson Jr., and Garret M. Hampton. Analysis of gene expression identifies candidate markers and pharmacological targets in prostate cancer. 61(16):5974–5978, 2001.

[20] Guangchuang Yu, Fei Li, Yide Qin, Xiaochen Bo, Yibo Wu, and Shengqi Wang. GOSemSim: an R package for measuring semantic similarity among GO terms and gene products. *Bioinformatics*, 26(7):976–978, 2010.

Virchows Arch (2011) 458:323–330
DOI 10.1007/s00428-010-1018-1

ORIGINAL ARTICLE

PTOV1 is overexpressed in human high-grade malignant tumors

Sara Fernández · Jose L. Mosquera · Lide Alaña · Alex Sanchez-Pla · Juan Morote · Santiago Ramón y Cajal · Jaume Reventós · Inés de Torres · Rosanna Paciucci

Received: 14 October 2010 / Revised: 22 November 2010 / Accepted: 23 November 2010 / Published online: 23 December 2010

Abstract The prostate tumor overexpressed-1 (PTOV1) protein was first described overexpressed in prostate cancer but not detected in normal prostate. PTOV1 expression is associated to increased cancer proliferation in vivo and in vitro. In prostate biopsy, PTOV1 detection is helpful in the early diagnosis of cancer. The purpose of this study was to analyze the relevance of PTOV1 expression to identify aggressive tumors derived from 12 different histological tissues. Tissue microarrays (TMAs) containing 182 biopsy samples, including 168 human tumors, were analyzed for PTOV1 and Ki67 expression by immunohistochemistry. Tumors of low and high histological grade were selected from lung, breast, endometrium, pancreas liver, skin, ovary, colon, stomach, kidney, bladder, and cerebral gliomas. One TMA with representative tissues without cancer (14 samples) was used as control. PTOV1 expression was analyzed semiquantitatively for the intensity and percentage of positive cells. Ki67 was evaluated for tumors proliferative index. Results show that PTOV1 was expressed in over 95% of tumors examined. Its expression was significantly associated to high-grade tumors (p=0.014). This association was most significant in urothelial bladder carcinomas (p=0.026). Overall, the expression of Ki67 was associated to high-grade tumors, and it was significant in several tumor types. PTOV1 and Ki67 were significantly co-overexpressed in all tumors (p=0.001), and this association was significant in clear cell renal carcinoma (p=0.005). In conclusion, PTOV1 expression is associated to more aggressive human carcinomas and more significantly to bladder carcinomas suggesting that this protein is a potential new marker of aggressive disease in the latter tumors.

Keywords PTOV1 · Immunohistochemistry · Human tumors · Low and high grade of malignancy · Bladder cancer · Renal carcinoma

Introduction

Prostate tumor overexpressed-1 (PTOV1) was identified as a novel gene and protein during a differential display

Inés de Torres and Rosanna Paciucci equally contributed to the work.

S. Fernández · S. Ramón y Cajal · I. de Torres
Department of Pathology, Vall Hebron Hospital,
Barcelona, Spain

A. Sanchez-Pla
University of Barcelona,
Barcelona, Spain

J. Morote
Department of Urology, Vall Hebron Hospital,
Barcelona, Spain

L. Alaña · J. Reventós
Research Unit in Biomedicine and Translational and Pediatrics
Oncology, Vall Hebron Institute of Research,
Colserola building, Pg. Vall d'Hebron 119-129,
Barcelona 08035, Spain

J. L. Mosquera · A. Sanchez-Pla
Unit of Statistic–Bioinformatic, Vall Hebron Institute of Research,
Barcelona, Spain

J. Morote · S. Ramón y Cajal · I. de Torres
Autonoma University,
Barcelona, Spain

R. Paciucci (✉)
Research Unit in Biomedicine and Translational and Pediatrics
Oncology, Vall Hebron Institute of Research,
Colserola building, Pg. Vall d'Hebron 119-129,
Barcelona 08035, Spain
e-mail: rpaciucci@ir.vhebron.net

The Jejunum of Diarrhea-Predominant Irritable Bowel Syndrome Shows Molecular Alterations in the Tight Junction Signaling Pathway That Are Associated With Mucosal Pathobiology and Clinical Manifestations

Cristina Martínez, PhD[1], María Vicario, PhD[1,2], Laura Ramos, MD[1], Beatriz Lobo, MD[1], Jose Luis Mosquera, BSc[3], Carmen Alonso, MD, PhD[1], Alex Sánchez, PhD[3,4], Mar Guilarte, MD[1,5], María Antolín, PhD[1,2], Inés de Torres, MD, PhD[6], Ana M. González-Castro, PhD[1], Marc Pigrau, MD[1], Esteban Saperas, MD, PhD[1], Fernando Azpiroz, MD, PhD[1,2] and Javier Santos, MD, PhD[1,2]

OBJECTIVES: **Diarrhea-predominant irritable bowel syndrome (IBS-D) patients show altered epithelial permeability and mucosal micro-inflammation in both proximal and distal regions of the intestine. The objective of this study was to determine the molecular events and mechanisms and the clinical role of upper small intestinal alterations.**

METHODS: **Clinical assessment and a jejunal biopsy was obtained in IBS-D patients and healthy subjects. Routine histology and immunohistochemistry was performed in all participants to assess the number of mast cells (MCs) and intraepithelial lymphocytes. RNA in tissue samples was isolated to identify genes showing consistent differential expression by microarray analysis followed by pathway and network analysis in order to identify the biological functions of the differentially expressed genes in IBS-D. Gene and protein expression of tight junction (TJ) components was also assessed by quantitative real-time polymerase chain reaction and confocal microscopy to evaluate the pathways identified by gene expression analysis.**

RESULTS: **The analysis reveals a strong association between the transcript signature of the jejunal mucosa of IBS-D and intestinal permeability, MC biology, and TJ signaling. The expression of zonula occludens 1 (ZO-1) was reduced in IBS-D at both gene and protein level, with protein redistribution from the TJ to the cytoplasm. Remarkably, our analysis disclosed significant correlation between ZO proteins, MC activation, and clinical symptoms.**

CONCLUSIONS: **IBS-D manifestations are linked to molecular alterations involving MC-related dysregulation of TJ functioning in the jejunal mucosa.**

SUPPLEMENTARY MATERIAL is linked to the online version of the paper at http://www.nature.com/ajg

Am J Gastroenterol advance online publication, 13 March 2012; doi:10.1038/ajg.2011.472

INTRODUCTION

There is convincing epidemiological evidence to support the relation between psychosocial determinants (1–3), gastrointestinal infections (4), and the onset, persistence, and severity of irritable bowel syndrome (IBS) manifestations. However, a pathophysiological connection has not been elucidated. Our general hypothesis is that a leading event in IBS pathogenesis is the breakdown of intestinal epithelial barrier's surveillance. This is inferred from

[1]Department of Gastroenterology, Digestive System Research Unit, Hospital Universitari Vall d'Hebron, Institut de Recerca Vall d'Hebron, Universitat Autònoma de Barcelona (Departmento de Medicina), Barcelona, Spain; [2]Centro de Investigación Biomédica en Red de Enfermedades Hepáticas y Digestivas (CIBERehd), Barcelona, Spain; [3]Statistics and Bioinformatics Unit, Institut de Recerca Vall d'Hebron, Barcelona, Spain; [4]Statistics Department, Facultad de Biología, Universidad de Barcelona, Barcelona, Spain; [5]Department of Allergy, Hospital Universitari Vall d'Hebron, Universitat Autònoma de Barcelona (Departmento de Medicina), Barcelona, Spain; [6]Department of Pathology, Hospital Universitari Vall d'Hebron, Universitat Autònoma de Barcelona (Departmento de Medicina), Barcelona, Spain. **Correspondence:** Javier Santos, MD, PhD, Department of Gastroenterology, Digestive System Research Unit, Laboratory of Neuro-Immuno-Gastroenterology, Institut de Recerca Vall d'Hebron and University Hospital Vall d'Hebron, Paseo Vall d'Hebron 119-129, 08035 Barcelona, Spain. E-mail: jsantos@ir.vhebron.net

Received 4 April 2011; accepted 4 December 2011

Analysis of Serum miRNA Profiles of Myasthenia Gravis Patients

Gisela Nogales-Gadea[1,2]*, Alba Ramos-Fransi[1,2], Xavier Suárez-Calvet[1,2], Miquel Navas[1,2], Ricard Rojas-García[1,2], Jose Luis Mosquera[3], Jordi Díaz-Manera[1,2], Luis Querol[1,2], Eduard Gallardo[1,2], Isabel Illa[1,2]*

1 Neuromuscular Diseases Unit, Hospital de la Santa Creu i Sant Pau, Universitat Autonoma de Barcelona, Barcelona, Spain, 2 CIBER de enfermedades neurodegenerativas (CIBERNED), Instituto de Salud Carlos III, Madrid, Spain, 3 Department of Statistics, University of Barcelona, Barcelona, Spain

Abstract

Myasthenia gravis (MG) is an autoimmune disease characterized by the presence of autoantibodies, mainly against the acetylcholine receptor (AChR). The mechanisms triggering and maintaining this chronic disease are unknown. MiRNAs are regulatory molecules that play a key role in the immune system and are altered in many autoimmune diseases. The aim of this study was to evaluate miRNA profiles in serum of 61 AChR MG patients. We studied serum from patients with early onset MG (n = 22), late onset MG (n = 27) and thymoma (n = 12), to identify alterations in the specific subgroups. In a discovery cohort, we analysed 381 miRNA arrays from 5 patients from each subgroup, and 5 healthy controls. The 15 patients had not received any treatment. We found 32 miRNAs in different levels in MG and analysed 8 of these in a validation cohort that included 46 of the MG patients. MiR15b, miR122, miR-140-3p, miR185, miR192, miR20b and miR-885-5p were in lower levels in MG patients than in controls. Our study suggests that different clinical phenotypes in MG share common altered mechanisms in circulating miRNAs, with no additional contribution of the thymoma. MG treatment intervention does not modify the profile of these miRNAs. Novel insights into the pathogenesis of MG can be reached by the analysis of circulating miRNAs since some of these miRNAs have also been found low in MG peripheral mononuclear cells, and have targets with important roles in B cell survival and antibody production.

Citation: Nogales-Gadea G, Ramos-Fransi A, Suárez-Calvet X, Navas M, Rojas-García R, et al. (2014) Analysis of Serum miRNA Profiles of Myasthenia Gravis Patients. PLoS ONE 9(3): e91927. doi:10.1371/journal.pone.0091927

Editor: Guher Saruhan-Direskeneli, Istanbul University, Turkey

Received November 15, 2013; **Accepted** February 17, 2014; **Published** March 17, 2014

Funding: This project was supported with a research grant from Fondo de Investigaciones Sanitarias (Fondo de Investigaciones [FIS] Intrasalud; 09/1964; principal investigator, I.I.). GNG and LQ are supported by FIS CD10/00027 and FIS CM09/00017 grants from Fondo de Investigaciones Sanitarias, Instituto de Salud Carlos III, Spain. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: Gisela Nogales-Gadea is editor of PLoS ONE. This do not alter the authors' adherence to PLOS ONE editorial policies and criteria.

* E-mail: gnogales@santpau.cat (GNG); iilla@santpau.cat (II)

Introduction

Myasthenia gravis (MG) is an autoimmune disease leading to fluctuating muscle weakness and fatigability. Patients with MG have been reported to have autoantibodies to the acetylcholine receptor (AChR), to MuSK or to LRP4 proteins [1,2,3]. Most MG patients, however, have circulating antibodies to AChR [4]. These antibodies are of the IgG subtype and their synthesis requires interaction between activated T and B cells [5]. Suggested mechanisms leading to autoantibody production include errors in antigen presentation or recognition [6,7,8], tolerance against self-antigens [9], and proliferation/apoptosis regulation of these immune cells [10,11].

MG patients with AChR antibodies are clinically heterogeneous [12]. Age at onset varies, and patients can be divided into early onset MG (EOMG), when symptoms appear before 50 years of age, or in late onset MG (LOMG), when they appear after 50 years [13]. Thymic involvement is also variable, more than 80% of EOMG patients have thymic hyperplasia [14] and 10–15% of MG patients have thymoma [15]. Thymectomy is used as a therapeutical intervention in EOMG [16] and in patients with thymoma. Response to treatment is also diverse. Most patients respond to steroids or other immunesuppressors, but some patients are refractory to standard therapy [15]. The heterogeneity is not only clinical and therapeutic. It may also involve the AChR antibody titers, which may be high or low independently of the patient's clinical status [17]. These findings suggest that the pathogenic mechanisms involved in each patient subgroup are different. No biomarkers are available, however, to predict such heterogeneity.

MiRNAs are small, non-coding regulatory molecules that modify gene expression by binding to the 3′ untranslated region of their target messenger RNAs [18]. These molecules are key in several cellular functions, and changes in their expression patterns have been associated with several diseases [19,20,21,22]. miRNAs play a diverse role in the immune system, participating in immune cell development, germinal center response, generation of Ig class-switched plasma cells, and response to toll-like receptor stimulus [23]. All of these mechanisms are potentially involved in the development of AChR antibodies. MiRNA expression profiles have been previously studied in peripheral blood mononuclear cells of MG patients [24,25] and let-7c and miR320 have been found downregulated. Functional studies have shown that these two miRNAs can contribute to MG induction or progression by regulating the expression of some cytokines. A recent study has shown that miR146a is upregulated in patients, and it can be regulating genes as CD40, CD80, TLR4 and NFkB [26].

Circulating miRNAs have been extensively studied from their discovery [27,28], as they have been found altered in different

Appendix B

Examples Associated with Semantic Similarities

In this appendix some examples associated with chapter of *Material and Methods* 3 of second part of the thesis II are illustrated just for pedagogical purposes. These examples are presented in order of appearance in the text with the corresponding section reference.

B.1 Examples of Basic Graph Concepts Section

Example B.1. *Figure B.1 depicts a graph* $G_1 = (V_1, E_1)$ *where* $V_1 = \{v_1, v_2, v_3, v_4, v_5, v_6\}$ *and* $E_1 = \{e_{12}, e_{13}, e_{23}, e_{24}, e_{36}, e_{56}\}$

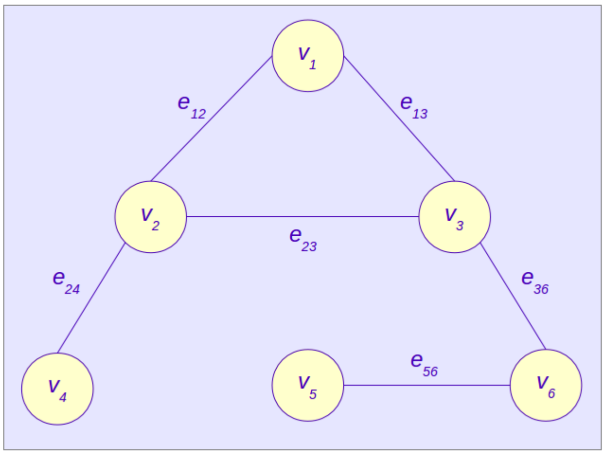


Figure B.1: Representation of graph $G_1 = (V_1, E_1)$.

Example B.2. *Figure B.1 depicts a multigraph* $G_2 = (V_1, E_2)$ *where* $V_1 = \{v_1, v_2, v_3, v_4, v_5, v_6\}$ *and* $E_2 = \{e_{12}, e_{13}, e_{23}, e_{23}, e_{24}, e_{36}, e_{44}, e_{56}\}$

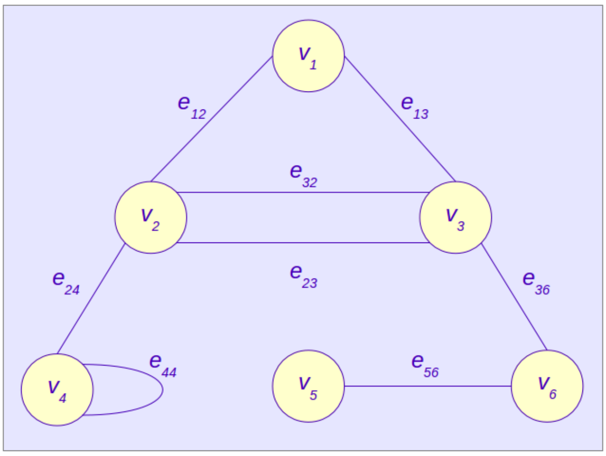


Figure B.2: Representation of multigraph $G_2 = (V_1, E_2)$.

B.2 Example of Subgraphs Section

Example B.3. *Figure B.3 shows a subgraph* S_1 *from the graph* G_1 *(see example B.1) induced by* $V_{S_1} = \{v_1, v_2, v_3, v_4, v_5\}$ *and* $E_{S_1} = \{e_{12}, e_{13}, e_{24}\}$*.*

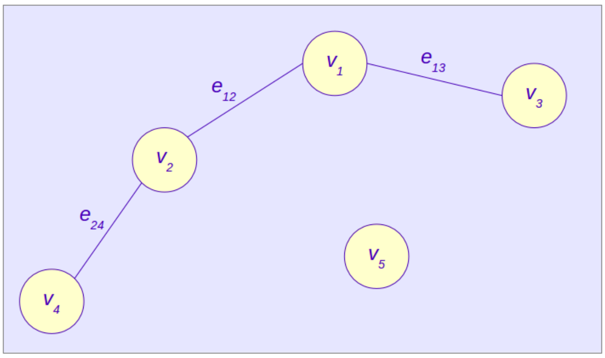


Figure B.3: Representation of subgraph S_1.

B.3 Example of Directed Graphs Section

Example B.4. *Consider the following incidence function that defines a set of "directions" on the undirected graph* G_2

$$\begin{array}{rccl} \psi : & E_2 & \longrightarrow & V_2 \times V_2 \\ & e_{12} & \longmapsto & \psi_{12} \\ & e_{13} & \longmapsto & \psi_{13} \\ & e_{23} & \longmapsto & \psi_{23} \\ & e_{24} & \longmapsto & \psi_{24} \\ & e_{32} & \longmapsto & \psi_{32} \\ & e_{36} & \longmapsto & \psi_{36} \\ & e_{44} & \longmapsto & \psi_{44} \\ & e_{56} & \longmapsto & \psi_{56} \end{array}$$

So, by applying this incidence function, a digraph $D_1 = (V_2, E_2)$ *arises from it*

$$\psi(E_2) = \{\psi_{12}, \psi_{13}, \psi_{23}, \psi_{24}, \psi_{32}, \psi_{36}, \psi_{44}, \psi_{56}\}.$$

Figure B.4 shows the associated representation.

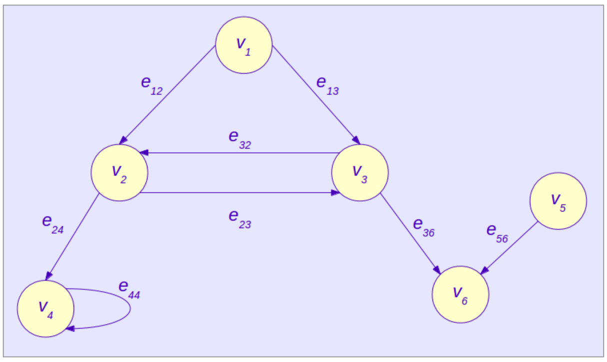


Figure B.4: Representation of digraph D_1.

Notice that D_1 is an orientation of G_2.

B.4 Example of Paths and Connection Section

Example B.5. *In the directed graph D_1, $P = (v_1, v_2, v_3, v_6)$ is a path between the origin v_1 and the terminus v_6, and v_2, v_3 are internal nodes. That is, v_6 is reachable from v1, and this path consist of three arcs e_{12}, e_{23}, e_{36}. Thus, we say that $P = P^3$ is a path of length 3.*

B.5 Example of DAG and Rooted DAG Section

Example B.6. *Let ψ be an incidence function applied on the undirected graph G_1 such that*

$$\begin{array}{rccc} \psi : & E_3 & \longrightarrow & V_1 \times V_1 \\ & e_{ij} & \longmapsto & \psi_{ij} \end{array}$$

where $E_3 = \{e_{12}, e_{13}, e_{24}, e_{32}, e_{36}, e_{56}\}$ and $V_1 = \{v_1, v_2, v_3, v_4, v_5, v_6\}$. Then, the resulting digraph D_2 is a DAG whose representation is shown in figure B.5

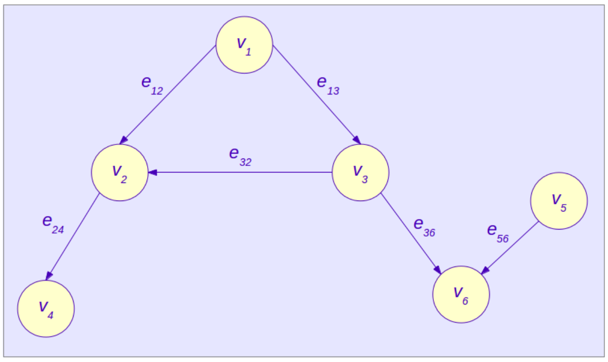


Figure B.5: Representation of DAG $D_2 = (V_1, E_3)$.

B.6 Examples of Matrices and Graphs Section

Example B.7. *Consider DAG D_2 (see example B.6). Then, its associated adjacency matrix is*

$$\begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \end{array}\begin{array}{c} \begin{array}{cccccc} v_1 & v_2 & v_3 & v_4 & v_5 & v_6 \end{array} \\ \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array} = \mathbf{A}_{D_2}$$

Example B.8. *The incidence matrix associated with DAG D_2 (see example B.6) is*

$$\begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \end{array}\begin{array}{c} \begin{array}{cccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 \end{array} \\ \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & -1 & 0 & 0 \\ 0 & -1 & 0 & 1 & 1 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -1 & -1 \end{pmatrix} \end{array} = \mathbf{B}_{D_2}$$

Example B.9. *Consider the adjacency matrix $\mathbf{A}_{D_2}$ (see example B.7).*

v	$d^-(v)$	$d^+(v)$	$d(v)$
v_1	0	2	2
v_2	2	2	4
v_3	2	2	4
v_4	2	1	3
v_5	0	1	1
v_6	2	0	2

Table B.1: Degree measures for each node v in digraph D_1.

Then, its accessibility matrix is

$$\mathbf{R}_{D_2} = Id_6 \oplus \mathbf{A}_{D_2} \oplus \ldots \oplus \mathbf{A}^5_{D_2} = \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

B.7 Examples of Order and Degrees Section

Example B.10. *The order of graphs G_1 and G_2 is $|G_1| = |G_2| = 6$, but while the degree for node v_3 in graph G_1 is $d(v_3) = 3$ and in graph G_2 is $d(v_3) = 4$, because their adjacent nodes are v_1, v_2, v_6 and v_1, v_2, v_2, v_6, respectively.*

Example B.11. *Consider digraph D_1. Then, its average degree is*

$$d(D_1) = \frac{2+4+4+3+1+2}{6} = 2,\widehat{6} \simeq 3.$$

and the different degree measures for each node in D_1 are shown in table B.1

B.8 Examples of Refinement of Relationships Section

Example B.12. *Let $\mathcal{T} = \{1, B, C, \ldots, D\}$ be a set of 12 terms in a small ontology described in figure B.6.*

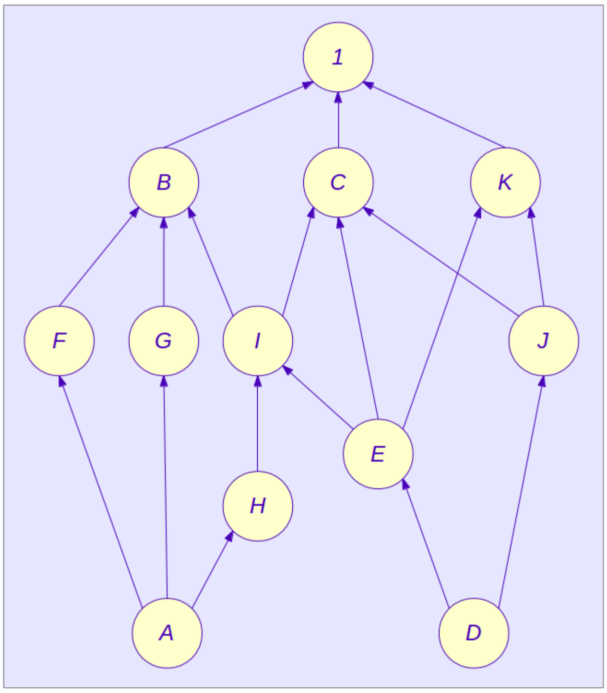


Figure B.6: DAG of a small ontology with 12 terms

The associated refinement matrix of this rooted DAG is

$$\begin{array}{cc}
 & \begin{array}{cccccccccccc} 1 & B & C & K & F & G & I & E & J & H & A & D \end{array} \\
\begin{array}{c} 1 \\ B \\ C \\ K \\ F \\ G \\ I \\ E \\ J \\ H \\ A \\ D \end{array} &
\begin{pmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0
\end{pmatrix}
\end{array} = \Gamma.$$

Example B.13. *The accessibility matrix associated with DAG depicted in figure B.6 is*

$$\begin{array}{c|cccccccccccc}
 & 1 & B & C & K & F & G & I & E & J & H & A & D \\
\hline
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
B & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
C & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
K & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
F & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
G & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
I & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
E & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
J & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
H & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
A & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\
D & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0
\end{array} = \mathbf{A}.$$

B.9 Examples of Mapping Genes to GO Section

Example B.14. *Let $\Omega = \{a, b, c, \ldots, j\}$ be a list of ten objects that are annotated in the ontology depicted in example B.12. Figure B.7 shows the Object-Ontology Complex associated with the relation between the objects and the ontology terms. This relation is made by mapping each object to the most refined term. For instance, object b maps to terms A, F and E. Thus, this object b is associated with the mentioned terms and its ancestors, but neither with term D because it is a refinement of term E, nor term J because it is not a target of b.*

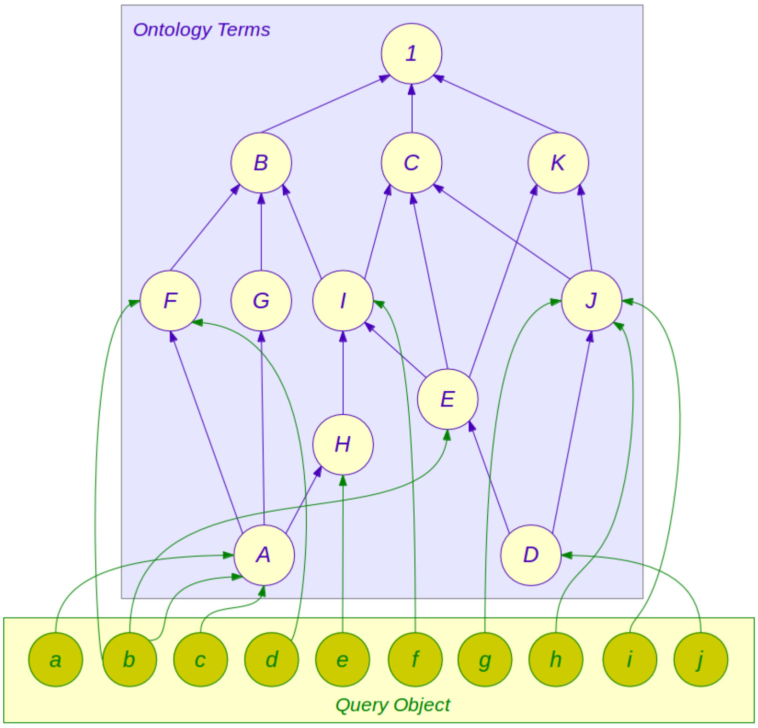


Figure B.7: Representation of an OOC with 10 objects annotated in an ontology with 12 terms.

Example B.15. *The mapping matrix of the OOC depicted in figure B.7 is*

$$\begin{array}{c|cccccccccccc}
 & 1 & B & C & K & F & G & I & E & J & H & A & D \\
\hline
a & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
b & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\
c & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
d & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
e & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
f & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
g & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
h & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
i & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
j & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{array} = \mathbf{M}.$$

Example B.16. *The coverage matrix associated with the Object-Ontology Complex depicted in figure B.7 is*

$$\begin{array}{c|cccccccccccc}
 & 1 & B & C & K & F & G & I & E & J & H & A & D \\
\hline
a & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\
b & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 \\
c & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\
d & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
e & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\
f & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
g & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
h & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
i & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
j & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1
\end{array} = \mathbf{C}.$$

B.10 Example of POSET Ontology Section

Example B.17. *The POSO structure $\mathcal{O} = \langle \mathcal{P}, X, F \rangle$ and its labelled poset associated with figure B.7 can be defined by assuming that $\mathcal{P}$ is a finite poset,*

$X = \{a, b, \ldots, j\}$ is the set of labels, and

$$\begin{array}{rccl} F: & X & \longrightarrow & 2^P \\ & a & \mapsto & F(a) = \{A\} \\ & b & \mapsto & F(b) = \{A, E, F\} \\ & c & \mapsto & F(c) = \{A\} \\ & d & \mapsto & F(d) = \{F\} \\ & e & \mapsto & F(e) = \{H\} \\ & f & \mapsto & F(f) = \{I\} \\ & g & \mapsto & F(g) = \{J\} \\ & h & \mapsto & F(h) = \{J\} \\ & i & \mapsto & F(i) = \{J\} \\ & j & \mapsto & F(j) = \{D\} \end{array}$$

is the the "mapping" function.

B.11 Examples of Lord's Measure Section

Example B.18. *Table B.2 shows a summary about the Information Content associated with each term of the Object-Ontology Complex depicted in example B.14*

t	$n(t)$	$P(t)$	$i(t)$
1	34	1.00000000	0.0000000
B	15	0.44117647	0.8183103
C	13	0.38235294	0.9614112
K	6	0.17647059	1.7346011
F	5	0.14705882	1.9169226
G	3	0.08823529	2.4277482
I	7	0.20588235	1.5804504
E	2	0.05882353	2.8332133
J	4	0.11764706	2.1400662
H	4	0.11764706	2.1400662
A	3	0.08823529	2.4277482
D	1	0.02941176	3.5263605

Table B.2: IC's of each term in the OOC depicted in figure B.7

Example B.19. *Table B.3 shows a matrix where each element is the Resnik's similarity for each pair of terms of the ontology depicted in figure B.7.*

	1	*B*	*C*	*K*	*F*	*G*	*I*	*E*	*J*	*H*	*A*	*D*
1	0											
B	0	0										
C	0	0	0									
K	0	0	0	0								
F	0	0.818	0	0	0							
G	0	0.818	0	0	0.818	0						
I	0	0.818	0.961	0	0.818	0.818	0					
E	0	0.818	0.961	1,735	0.818	0.818	1.580	0				
J	0	0	0.961	1,735	0	0	0.961	1.735	0			
H	0	0.818	0.961	0	0.818	0.818	1.580	1.580	0.961	0		
A	0	0.818	0.961	0	1.917	2.428	1.580	1.580	0.961	2.140	0	
D	0	0.818	0.961	1,735	0.818	0.818	1.580	2.833	2.140	1.580	1.580	0

Table B.3: Measures calculated with the semantic similarity of Resnik between all terms of the OOC depicted in figure B.7

B.11.1 Example of Joslyn's Measure Section

Example B.20. *Table B.4 shows the matrix with the minimum chain length between each pair of terms in the Object-Ontology Complex. Note that, these values are equal to the length of the shortest path if and only if terms are comparable. Non-comparable term are indicated with hyphen.*

	1	*B*	*C*	*K*	*F*	*G*	*I*	*E*	*J*	*H*	*A*	*D*
1	0											
B	1	0										
C	1	-	0									
K	1	-	-	0								
F	2	1	-	-	0							
G	2	1	-	-	-	0						
I	2	1	1	-	-	-	0					
E	2	2	1	1	-	-	1	0				
J	2	-	1	1	-	-	-	-	0			
H	3	2	2	-	-	-	1	-	-	0		
A	3	2	3	-	1	1	2	-	-	1	0	
D	3	3	2	2	-	-	2	1	1	-	-	0

Table B.4: Minimum chain length between each pair of terms in the OOC depicted in figure B.7

In order to understand better how this pseudo-distance matrix is built, the computation of the minimum chain length between comparable nodes D and K is illustrated. To calculate

$$\delta_m(D,K) = h_*(D,K) = \min_{C\in\mathcal{C}(D,K)} |C|$$

the set of all chains between nodes D and $K, \mathcal{C}(D,K)$ is required. So, elements of such a set are:

$$\mathcal{C}(D,K) = \{\{D,E,K\},\{D,J,K\}\} = \{C_1,C_2\} \Rightarrow l(C_1) = l(C_2) = 2$$

Therefore, the minimum chain length between D and K is $\delta_m(D,K) = 2$.

B.12 Example Associated with the Results Section of the Information Content Concept

Example B.21. *Consider the OOC depicted in figure B.7. Then, the associated mapping matrix is*

$$\begin{array}{c|cccccccccccc}
 & 1 & B & C & K & F & G & I & E & J & H & A & D \\
\hline
a & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
b & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\
c & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
d & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
e & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
f & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
g & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
h & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
i & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
j & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{array} = \mathbf{M}.$$

The matrix with the number of paths of any length between each pair of terms is $\mathbf{I} + \Gamma + \Gamma^2 + \ldots + \Gamma^r$, where r is the depth of the ontology, that is, the length of the longest path from the root term to the most specific refinement.

In this example $r = 4$. Thus,

$$\begin{array}{cc}
 & \begin{array}{cccccccccccc} 1 & B & C & K & F & G & I & E & J & H & A & D \end{array} \\
\begin{array}{c} 1 \\ B \\ C \\ K \\ F \\ G \\ I \\ E \\ J \\ H \\ A \\ D \end{array} &
\begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
2 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
4 & 1 & 2 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\
2 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
2 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\
4 & 3 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\
6 & 1 & 3 & 2 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1
\end{pmatrix}
\end{array} = \mathbf{M}(\mathbf{I}+\Gamma+\Gamma^2+\Gamma^3+\Gamma^4).$$

The product of these two matrices allows to compute a matrix that each element of it is the number of times that a term $t \in (t)$ has been referenced by both itself (column) and the refinements of an specific descendant (row). For instance, term K has been referenced 5 times. This is, because object j references term D, from this term to its ancestor K there are 2 paths, and objects g, h, and i references each of them term J that is a 1-step refinement of term F.

Now, if $N_t = \mathbf{M}(\mathbf{I}+\Gamma+\Gamma^2+\Gamma^3+\Gamma^4)$ is the matrix of the number of times that each term t_i or any of its specializations references to an specific ancestor t_j, just by summing the columns of such a matrix, $n(t)$'s associated with each term $t_j \in (T)$ is calculated (see table B.5).

t	$n(t)$
1	34
B	15
C	13
K	6
F	5
G	3
I	7
E	2
J	4
H	4
A	3
D	1

Table B.5: Number of times that each term t_j or any of its refinements appears in the OOC depicted in figure B.7

B.13 Example Associated with the Results Section of Lord's Measures

Example B.22. *Consider the Object-Ontology Complex described in figure B.7. If we are interested in estimating the semantic similarity between terms* E *an* J*, then* $S(E,J) = \{K, C, 1)\}$ *and*

$$\begin{aligned} sim_{Res}(E,J) &= \max\{i(K), i(C), i(1)\} \\ &= \max\{1.7346011, 0.9614112, 0.0000000\} \\ &= 1.7346011 \\ &= \min\{0.17647059, 0.38235294, 1.00000000\} \\ &= \min\{P(K), P(C), P(1)\} = \\ &= -\log P_{ms}(E,J) \\ &= sim_{Lord}(E,J). \end{aligned}$$