Advances in Intelligent Systems and Computing 477

Mohd Saberi Mohamad
Miguel P. Rocha
Florentino Fdez-Riverola
Francisco J. Domínguez Mayo
Juan F. De Paz *Editors*

10th International Conference on Practical Applications of Computational Biology & Bioinformatics

Springer

Advances in Intelligent Systems and Computing

Volume 477

Series editor

Janusz Kacprzyk, Polish Academy of Sciences, Warsaw, Poland
e-mail: kacprzyk@ibspan.waw.pl

About this Series

The series "Advances in Intelligent Systems and Computing" contains publications on theory, applications, and design methods of Intelligent Systems and Intelligent Computing. Virtually all disciplines such as engineering, natural sciences, computer and information science, ICT, economics, business, e-commerce, environment, healthcare, life science are covered. The list of topics spans all the areas of modern intelligent systems and computing.

The publications within "Advances in Intelligent Systems and Computing" are primarily textbooks and proceedings of important conferences, symposia and congresses. They cover significant recent developments in the field, both of a foundational and applicable character. An important characteristic feature of the series is the short publication time and world-wide distribution. This permits a rapid and broad dissemination of research results.

Advisory Board

More information about this series at http://www.springer.com/series/11156

Mohd Saberi Mohamad · Miguel P. Rocha
Florentino Fdez-Riverola
Francisco J. Domínguez Mayo
Juan F. De Paz
Editors

10th International Conference on Practical Applications of Computational Biology & Bioinformatics

Springer

Editors
Mohd Saberi Mohamad
Faculty of Computing
Universiti Teknologi Malaysia
Johor
Malaysia

Francisco J. Domínguez Mayo
ETS Ingeniería Informática
University of Sevilla
Sevilla
Spain

Miguel P. Rocha
Departamento de Informática Campus of Gualtar
Universidade do Minho
Braga
Portugal

Juan F. De Paz
Departamento de Informática y Automática
Universidad de Salamanca
Salamanca
Spain

Florentino Fdez-Riverola
Edificio Politécnico. Despacho 408 Campus Universitario
Escuela Superior de Ingeniería Informática
Ourense
Spain

ISSN 2194-5357 ISSN 2194-5365 (electronic)
Advances in Intelligent Systems and Computing
ISBN 978-3-319-40125-6 ISBN 978-3-319-40126-3 (eBook)
DOI 10.1007/978-3-319-40126-3

Library of Congress Control Number: 2016940897

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG Switzerland

Preface

Biological and biomedical research are increasingly driven by experimental techniques that challenge our ability to analyse, process and extract meaningful knowledge from the underlying data. The impressive capabilities of next generation sequencing technologies, together with novel and ever evolving distinct types of omics data technologies, have put an increasingly complex set of challenges for the growing fields of Bioinformatics and Computational Biology. To address the multiple related tasks, for instance in biological modeling, there is the need to, more than ever, create multidisciplinary networks of collaborators, spanning computer scientists, mathematicians, biologists, doctors and many others.

The International Conference on Practical Applications of Computational Biology & Bioinformatics (PACBB) is an annual international meeting dedicated to emerging and challenging applied research in Bioinformatics and Computational Biology. Building on the success of previous events, the 8th edition of PACBB Conference will be held on 1–3 June 2016 in the University of Sevilla, Spain. In this occasion, special issues will be published by the Interdisciplinary Sciences-Computational Life Sciences, Journal of Integrative Bioinformatics, Neurocomputing, Journal of Computer Methods and Programs in Biomedicine, Knowledge and Information Systems: An International Journal covering extended versions of selected articles.

This volume gathers the accepted contributions for the 10th edition of the PACBB Conference after being reviewed by different reviewers, from an international committee composed of 79 members from 11 countries. PACBB'16 technical program includes 22 papers spanning many different sub-fields in Bioinformatics and Computational Biology.

Therefore, this event will strongly promote the interaction of researchers from diverse fields and distinct international research groups. The scientific content will be challenging and will promote the improvement of the valuable work that is being carried out by the participants. In addition, it will promote the education of young scientists, in a post-graduate level, in an interdisciplinary field.

We would like to thank all the contributing authors and sponsors, as well as the members of the Program Committee and the Organizing Committee for their hard and highly valuable work and support. Their effort has helped to contribute to the success of the PACBB'16 event. PACBB'16 wouldn't exist without your assistance.

June 2016

Mohd Saberi Mohamad
Miguel P. Rocha
Florentino Fdez-Riverola
Francisco J. Domínguez Mayo
Juan F. De Paz

Organization

General Co-chairs

Mohd Saberi Mohamad	Universiti Teknologi Malaysia
Miguel Rocha	University of Minho, Portugal
Florentino Fdez-Riverola	University of Vigo, Spain
Francisco José Domínguez Mayo	University of Sevilla, Spain

Program Committee

Alejandro F. Villaverde	IIM-CSIC, Spain
Alejandro Rodríguez	Polytechnic University of Madrid, Spain
Alexandre Perera	Universitat Politècnica de Catalunya, Spain
Alfonso Rodríguez	Universidad Politecnica de Madrid, Spain
Alfredo Vellido	Universitat Politècnica de Catalunya, Spain
Alicia Troncoso	University Pablo de Olavide, Spain
Amin Shoukry	Egypt-Japan University of Science and Technology, Egypt
Amparo Alonso	University of A Coruña, Spain
Ana Cristina Braga	University of Minho, Portugal
Ana Margarida Sousa	University of Minho, Portugal
Anália Lourenço	University of Vigo, Spain
Antonio Prestes	Universidad Politécnica de Madrid, Spain
Armando Pinho	Universty of Aveiro, Portugal
Boris Brimkov	Rice University, USA
Carole Bernon	IRIT/UPS, France
Carolyn Talcott	Stanford University, USA
Consuelo Gonzalo	Technical University of Madrid, Spain

Daniel Glez-Peña	University of Vigo, Spain
Daniela Correia	Centre of Biological Engineering, Portugal
David Hoksza	Charles University in Prague, Czech Republic
David Rodríguez	IIM-CSIC, Spain
Eduardo Valente	IPCB, Portugal
Eva Lorenzo	University of Vigo, Spain
Fernanda Brito Correia	DETI/IEETA University of Aveiro and DEIS/ISEC/Polytechnic Institute of Coimbra, Portugal
Fernando de la Prieta	University of Salamanca, Spain
Fernando Diaz-Gómez	University of Valladolid, Spain
Filipe Liu	University of Minho, Portugal
Gabriel Villarrubia	University of Salamanca, Spain
Gael Pérez	University of Vigo, Spain
Guillermo Calderón	Universidad Autónoma de Manizales, Colombia
Gustavo Isaza	University of Caldas, Colombia
Gustavo Santos-García	University of Salamanca, Spain
Hugo López-Fernández	University of Vigo, Spain
Javier Bajo	Polytechnic University of Madrid, Spain
Javier Prieto	University of Salamanca, Spain
Jeferson Arango	Universidad de Caldas, Colombia
João Ferreira	University of Lisboa, Portugal
João Manuel Rodrigues	University of Aveiro, Portugal
Jorge Vieira	IBMC, Porto, Portugal
José Antonio Castellanos	University of Salamanca, Spain
Jose Ignacio Requeno	University of Zaragoza, Spain
José Luis Oliveira	Universty of Aveiro, Portugal
José Manuel Colom	University of Zaragoza, Spain
Josep Gómez	Universitat Rovira I Virgili, Spain
Juan Ranea	Universidad de Málaga, Spain
Julio R. Banga	IIM-CSIC, Spain
Loris Nanni	University of Bologna, Italy
Lourdes Borrajo	University of Vigo, Spain
Luis F. Castillo	University of Caldas, Colombia
Luis M. Rocha	Indiana University, USA
Mª Araceli Sanchís	University of Carlos III, Spain
Manuel Álvarez	University of A Coruña, Spain
Marcelo Maraschin	Federal University of Santa Catarina, Florianopolis, Brazil
Marcos Martínez	University of A Coruña, Spain
Mark Thompson	Leiden University Medical Center, Netherlands
Martín Pérez-Pérez	University of Vigo, Spain

Miguel Reboiro	University of Vigo, Spain
Miriam Rubio	Spanish National Cancer Research Centre (CNIO), Spain
Narmer Galeano	Cenicafé, Colombia
Nuno A. Fonseca	EMBL-EBI, European Bioinformatics Institute, UK
Nuno F. Azevedo	University of Porto, Portugal
Óscar Dias	CEB/IBB, Universidade do Minho, Portugal
Pablo Chamoso	University of Salamanca, Spain
Patricia González	University of A Coruña, Computer Architecture Group (GAC), Spain
Paula Jorge	IBB - CEB Centre of Biological Engineering, Portugal
Pedro Ferrerira	Genome Bioinformatics Lab (GBL), CRG, Spain
Pedro Sernadela	University of Aveiro, Portugal
Pierpaolo Vittorini	University of L'Aquila, Italy
René Alquezar Mancho	UPC, Spain
Rita Ascenso	Polytecnic Institute of Leiria, Portugal
Rosalía Laza	University of Vigo, Spain
Rubén Romero	University of Vigo, Spain
Rui Camacho	University of Porto, Portugal
Sara Rodríguez	University of Salamanca, Spain
Sergio Matos	DETI/IEETA, Portugal
Thierry Lecroq	Univeristy of Rouen, France
Vanessa Maria Gervin	Federal University of Santa Catarina, Florianopolis, Brazil
Vera Afreixo	University of Aveiro, Portugal
Xavier Domingo Almenara	Rovira i Virgili University, Spain

Organising Committee

Carlos Arevalo Maldonado	University of Sevilla, Spain
Gustavo Aragon Serrano	University of Sevilla, Spain
Irene Barba	University of Sevilla, Spain
Miguel Ángel Barcelona Liédana	Technological Institute of Aragon, Spain
Juan Manuel Cordero Valle	University of Sevilla, Spain
Francisco José Domínguez Mayo	University of Sevilla, Spain
Juan Pablo Domínguez Mayo	University of Sevilla, Spain
Manuel Domínguez Muñoz	University of Sevilla, Spain
María José Escalona Cuaresma	University of Sevilla, Spain

José Fernández Engo	University of Sevilla, Spain
Laura García Borgoñón	Technological Institute of Aragon, Spain
Julian Alberto García García	University of Sevilla, Spain
Javier García-Consuegra Angulo	University of Sevilla, Spain
José González Enríquez	University of Sevilla, Spain
Tatiana Guardia Bueno	University of Sevilla, Spain
Andrés Jiménez Ramírez	University of Sevilla, Spain
Javier Jesús Gutierrez Rodriguez	University of Sevilla, Spain
Manuel Mejías Risoto	University of Sevilla, Spain
Laura Polinario	University of Sevilla, Spain
José Ponce Gonzalez	University of Sevilla, Spain
Francisco José Ramírez López	University of Sevilla, Spain
Isabel Ramos Román	University of Sevilla, Spain
Jorge Sedeño López	University of Sevilla, Spain
Nicolás Sánchez Gómez	University of Sevilla, Spain
Juan Miguel Sánchez Begines	University of Sevilla, Spain
Eva-Maria Schön	University of Sevilla, Spain
Jesús Torres Valderrama	University of Sevilla, Spain
Carmelo Del Valle Sevillano	University of Sevilla, Spain
Antonio Vázquez Carreño	University of Sevilla, Spain
Carlos Torrecilla Salinas	University of Sevilla, Spain
Ainara Aguirre Narros	University of Sevilla, Spain
Juan Antonio Alvarez García	University of Sevilla, Spain

PACBB 2016 Sponsors


UNIVERSIDADE
DE VIGO


Universidade do Minho
Instituto de Educação


indra


IEEE
SECCIÓN ESPAÑA


IEEE
IEEE Systems, Man and Cybernetics Society
Spain


Portuguese
Association for
Artificial
Intelligence

Contents

Part IV Systems Biology

Part I
Data and Text Mining

Intelligent Systems for Predictive Modelling in Cheminformatics: QSPR Models for Material Design Using Machine Learning and Visual Analytics Tools

F. Cravero, M.J. Martinez, G.E. Vazquez, M.F. Díaz and I. Ponzoni

Abstract In this paper, the use of intelligence systems for feature extraction in predictive modelling applied to Cheminformatics is presented. In this respect, the application of these methods for predicting mechanical properties related to the design of the polymers constitutes, by itself, a central contribution of this work, given the complexity of *in silico* studies of macromolecules and the few experiences reported in this matter. In particular, the methodology evaluated in this paper uses a features learning method that combines a quantification process of 2D structural information of materials with the autoencoder method. Several inferred models for *tensile strength at break*, which is a mechanical property of materials, are discussed. These results are contrasted to QSPR models generated by traditional approaches using accuracy metrics and a visual analytic tool.

Keywords Cheminformatics · QSPR · Polymeric materials · Machine learning · Visual analytics

F. Cravero · M.F. Díaz
Planta Piloto de Ingeniería Química, Universidad Nacional del Sur – CONICET, Bahía Blanca, Argentina

G.E. Vazquez
Facultad de Ingeniería y Tecnologías, Universidad Católica del Uruguay, Montevideo, Uruguay

M.J. Martinez · I. Ponzoni(✉)
Instituto de Ciencias e Ingeniería de la Computación, Universidad Nacional del Sur – CONICET, Bahía Blanca, Argentina
e-mail: ip@cs.uns.edu.ar

M.S. Mohamad et al. (eds.), *10th International Conference on PACBB*,
Advances in Intelligent Systems and Computing 477,
DOI: 10.1007/978-3-319-40126-3_1

1 Introduction

The development of intelligent systems with application to cheminformatics is an active field of research [1]. In this sense, the use of machine learning methods for the design of QSPR (Quantitative Structure-Property Relationship) models has been increasing significantly during last decades [1]. The inference of QSPR models constitutes a particular case of predictive modeling problem, where a domain expert is focused on discovering the relationship between several molecular descriptors (which characterizes the structure and other features of chemical compounds) and a target under study (related with a physicochemical or mechanical property of interest).

To address this predictive modeling problem using intelligent systems requires tackling several computer science subproblems. One of this is the selection of the most relevant molecular descriptors for the property under consideration. This is a typical feature selection problem, where the featured variables are molecular descriptors (quantitatively calculated) and the target variable is a physicochemical or mechanical property. In this cheminformatics domain, the usual number of candidate molecular descriptors to be consider for designing a QSPR model is huge (around thousands of molecular descriptors can be computed with commercial tools). Therefore, the automatic selection of an optimal set of molecular descriptors is a highly expensive task [2]. Moreover, the computational effort is even harder when the feature selection method is applied to the prediction of QSPR models in the design of new polymeric materials. In this specific cheminformatics subdomain, the size and characteristics of these chemical compounds (usually huge macromolecules) turn difficult the computation of the most straightforward molecular descriptors [3].

For this reason, several techniques of dimensionality reduction and feature extraction have emerged in the area of QSPR modeling to simplify, or even avoid, the selection of descriptors. An example of these approaches is based on the combination of the CODES and TSAR algorithms [4]. This methodology generates a set of new features, using neural networks for the extraction process. These new features configure a new space of variables, with reduced dimensionality, which captures information derived from the molecular structure of the compounds without need of run a descriptors calculation process followed of traditional features selection method.

CODES-TSAR (C-T) has been successfully applied to chemical compound databases related to pharmacological studies and other biological experiments [5], nonetheless their effectiveness has never been tested in the prediction of QSPR models for the design of synthetic polymers. Therefore, the main goal of this paper is to explore the scope and limitations of this methodology in the inference of mechanical properties relevant for Materials Science.

In particular, the QSPR models obtained by C-T are contrasted to QSAR models obtained by traditional QSPR design strategies using an integrative approach that uses machine learning and visual analytics tools. Some hybridization among the QSPR models obtained by the different strategies is also evaluated, in order to assess the impact of combining the predictive skills of these alternatives models.

2 Methodology

Currently, the methods based on QSPR related a property from molecular, structural and nonstructural descriptors that numerically quantify different aspects of a molecule [1]. In mathematical terms, a QSPR model is presented as a function $Y = f(X)$, where $X = (x_1, x_2, \ldots, x_n)$ is a chemical compound database represented as a vector of molecular descriptors and Y is an experimental target property.

The goal is to infer f from a dataset with chemical compounds, where a number of molecular descriptors are computed for each compound using specific tools like DRAGON [6]. Besides it is also required experimental data for the physicochemical property or biological activity of interest (Y). From this dataset by using training method, the function f is learned. Once f has been inferred, this function can be applied to the compounds not covered by training. Thus, f can predict in silico the value of a property based on the analysis of data from other experiments. To set this function f is necessary to identify, first, what are the molecular descriptors that are related to the property and, therefore, provide more information to the QSPR model.

A traditional approach to carry out this selection process is to perform a combinatorial search, for this we use the DELPHOS software. DELPHOS is an intelligent system that uses a wrapper multi-objective optimization method [7], based on the combination of different machine learning algorithms, which explores the space of possible subsets of descriptors and assess their predictive ability for estimating the value of the target property (Fig.1 - Right).

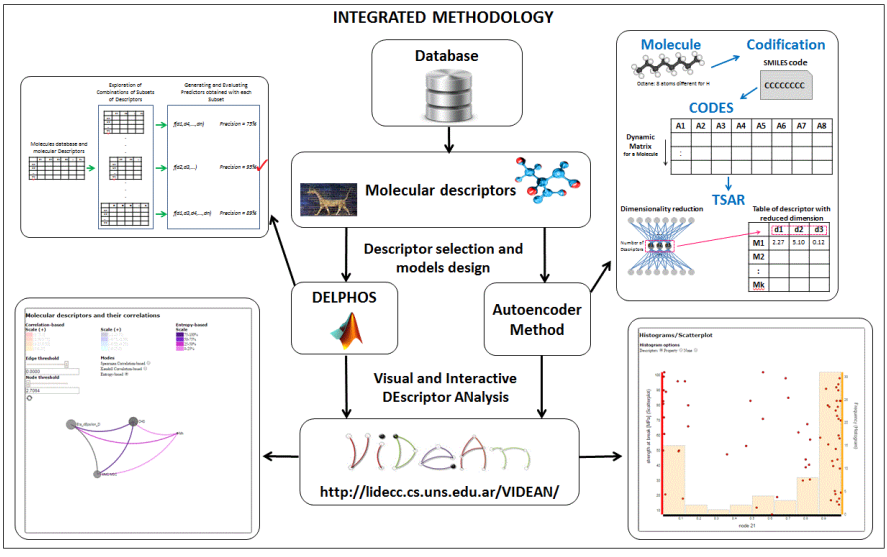


Fig. 1 Integrated Methodology: Traditional (in the right) vs. New Strategy (in the left)

An alternative to these traditional QSPR modeling methods is to use features learning techniques, such as the combination of the CODES and TSAR method [4]. This option avoids going through a process of combinatorial exploration required by the feature selection methods. With this methodology, given a database of compounds, the first step is to structurally describe the molecule through its SMILES code. As a second step, this code enters into the CODES tool, which generates a dynamic matrix with as many columns as the number of atoms (different than hydrogen) contains the molecule and as many rows as it is necessary for stabilization. Then, this matrix enters into the TSAR tool, responsible for performing dimensionality reduction by means of an autoencoder neural network [8]. Repeating this analysis for each molecule a reduced descriptor table is obtained (Fig 1. Left).

3 Experiments and Results

For this work, a database developed by our research group is used [9]. From this database 66 polymers were taken and characterized by their SMILES code. Polymers in the database are: linear, thermoplastic, amorphous, flame-retardant, thermally stable, hydrolytically stable, hydrolytic degradable, low toxic, and thermostable. And concerning international norms, database meets the following ones: ASTM D638, ASTM D882-83, and DIN 53504.53A. Also other characteristics were considered to make the database (Table 1).

Table 1 Ranges of values of the dataset polymers properties have the following

Properties	Ranges of values
Mn	4700 – 765000 [g/mol]
Mw	19500 – 2200000 [g/mol]
Mw/Mn	1.15 – 5.6
CHS	1 – 100 [mm/min]
Temperature	20 - 25 [ºC]
Tensile Strength at Break	7.5 – 103 [MPa]

The property selected for this study is *tensile strength at break*. Tensile strength comes from tensile test and it is the maximum effort that supports the test specimen during the experiment. When the maximum stress occurs at the yield point it is known as tensile strength at yield and the corresponding elongation is known as elongation at yield (Point B in Fig. 2). When the fracture occurs, the corresponding stress is known as *tensile strength at break* (Point D in Fig. 2). Materials which have a low *tensile strength at break* are often referred as weak materials.

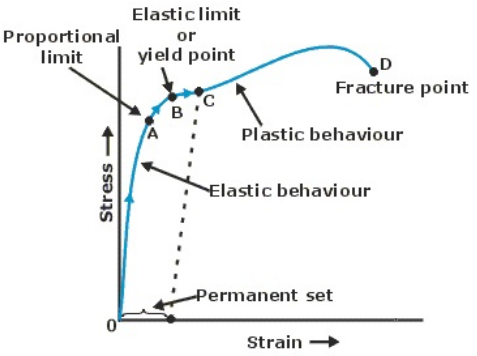


Fig. 2 A typical stress-strain curve for a polymer.

Based on the SMILE code for each monomer of the database, CODES get a dynamic matrix (record of the learned process), where each value represents the correlation between atoms, the atom bonds and connectivity with the rest of the molecule. The next step is the reduction of dimensions process for each molecule in order to have a reasonably small number of descriptors, by applying TSAR. Since this tool is based on an autoencoder algorithm, the number of nodes in the hidden middle layer determines the number of descriptors generated [8].

In previous reports [10] we experimented with two different network architectures, three and two neurons in the hidden layer, getting the models: C-T N3 and C-T N2. These models were compared with a model generated by using the classic methodology of our research team group: RM (Reference Model) [8]. RM consists of 4 descriptors and it was chosen by using a combination of a feature selection method and a physicochemical-motivated strategy: Number Average Molecular Weight (Mn), Cross-Head Speed (CHS), ETA_dEpsilon_D and the ratio of Mass of Main Chain by Mass of side chain (M_{MC}/M_{SC}). Also, combination of C-T N3 and C-T N2 including Mn y CHS (Enhanced Models) were tested, as well as whit RM (Combined Models). Thus, 7 different models were trained by using the combined sets of descriptors (Table 2). In particular, the CHS descriptor always determines the value of the mechanical properties because it is a testing parameter, and Mn provides "macro" information about the molecules.

Table 2 Models and Performance (Training and Validation metrics refer to R^2).

Models	Cardinality	Training	Validation
C-T N2	(2)	0.7071	0.5788
C-T N3	(3)	0.7513	0.4385
RM: Mn + CHS + Eta_dEpsilon_D + M_{MC}/M_{SC}	(4)	0.884	0.8172
Enhanced N2: C-T N2 + Mn + CHS	(4)	0.7954	0.7527
Enhanced N3: C-T N3 + Mn + CHS	(5)	0.8734	0.5857
Combined N2: C-T N2 + RM	(6)	**0.9333**	**0.8488**
Combined N3: C-T N3 + RM	(7)	**0.9253**	**0.8514**

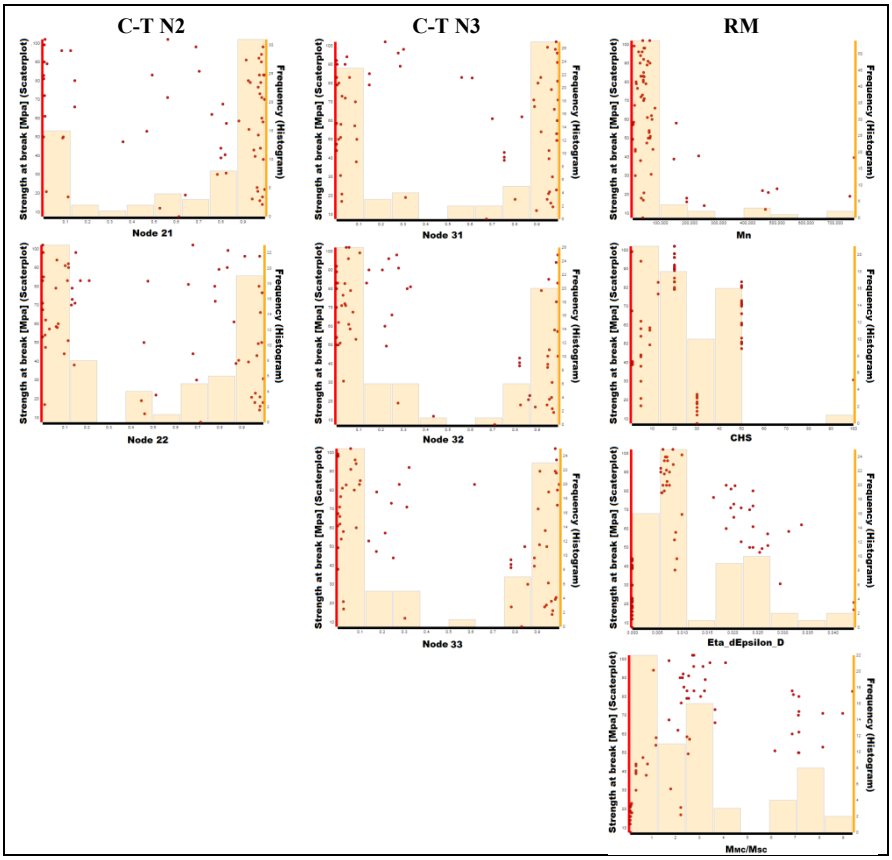


Fig. 3 Descriptors vs Property: Scatterplots and Histograms.

In general, a robust predictor should follow three fundamental principles, high predictive accuracy in statistical terms, low cardinality model (minimum number of descriptors) and high interpretability. Modeling polymers is complex but visual analytical computational tools as VIDEAN [2], simplifies the task of modeling and interpretability analysis. Statistical evaluation of the models [10] was performed with WEKA [11], using decision trees with 10-fold validation. The performances are shown in Table 2. The validation metrics of most of the models have reasonable predictive precision. It is necessary to note that although the cardinality of Combined and Enhanced Models is equal to or greater than the cardinality of the RM, this is not so relevant. As such, the set of descriptors obtained by the C-T method can be thought as a single descriptor (characterized by more than one value), since they must not be interpreted or used separately. In other words, the values generated by C-T together represent a single descriptor of the entire 2D structure. While it is appreciated that the models derived from the exclusive use of descriptors created by C-T do not reach the predictive performance of the RM; it appears that

the combination of the information provided by the C-T models and classical methodologies improves the statistical accuracy of the RM without a significant increase in cardinality.

At this point, VIDEAN was used to achieve a better understanding of the descriptors contributions to the different models. VIDEAN offers scatter plots of property vs. descriptor values (Fig. 3) and a representation of the degree of mutual information descriptors in a pairwise analysis (Fig. 4), while darker the purple color of the link is, more independent are the values between descriptors (lesser mutual information). The polymer database used in this study is available at htt://lidecc.cs.uns.edu.ar/VIDEAN.

Analyzing the model C-T N3 it is possible to see that the histograms of Fig. 3 show the same type of descriptor behavior respect to the property. This is, in general undesirable since it is expected that each descriptor provides information of different zones from the structure-property relationship. This could justify their low performance (0.4385, Table 2). For C-T N2, analysis and conclusions are similar. RM model has four descriptors with different behavior (Fig. 3), but we can observe an area with a lack of information at the left end of the plot (high values of descriptor); an area where both C-T models show values. This scenario could explain why the Combined Models increase their performance (0.8488 and 0.8514, Table 2).

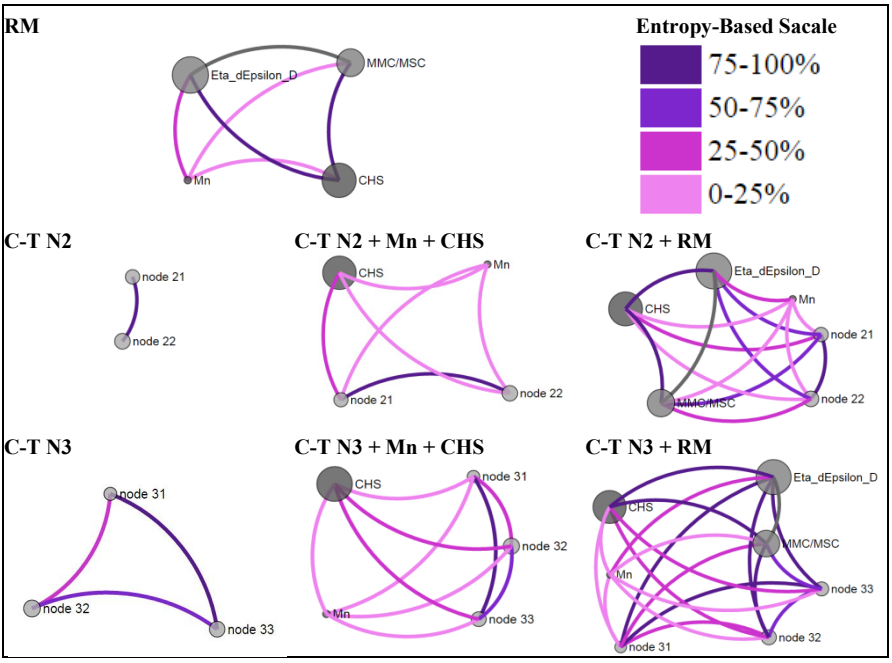


Fig. 4 Mutual information between descriptors of each model.

From Fig. 4 it can be concluded that in general all the descriptors have little mutual information except Mn, where all links are light pink. However, this descriptor is very significant for polymers since these materials do not have a single molecular weight but a molecular weight distribution. Mn is the number average molecular weight and gives macromolecular information to the model.

Thus, we can conclude that to model *tensile strength at break* using a database of polymeric materials the exclusive use of C-T method was not enough to predict the mechanical property. However, it was found that the new descriptors learned by this feature extraction technique can provide valuable information to QSPR models generated by other strategies.

4 Conclusions

In this paper, the use of feature learning techniques in QSPR modeling of polymeric materials was studied. C-T methodology, by means of an algorithm based on autoencoder, learns a reduced set of variables capturing structural 2D information of the molecule. The main objective was to establish if the quality of results achieved by the C-T approach for modeling of pharmacokinetic properties are preserved in QSPR models predicting oriented design of polymeric materials. With this goal, experiments were proposed to model the *tensile strength at break* in order to contrast the performance of the feature extraction technique against the development of a QSPR model based on expert knowledge (RM).

Regarding the models generated by C-T, it was found that the information captured by the inferred descriptors was not sufficient to describe the property. This led to the idea of extending the C-T models incorporating descriptors corresponding to macromolecular information and tensile test parameters. Furthermore, it was also decided to evaluate the performance of models that combine the descriptors generated by C-T with the RM. From these experiments, it was concluded that the new descriptors extracted by C-T provide information relevant to the RM. In the future it is planned to evaluate combined models for other mechanical properties of interest, such as elongation at break and tensile modulus, together with the use of other methods of machine learning.

Acknowledgments This work has been funded by grants PIP 11220120100471 and PIP 11420110100362.

References

1. Mitchell, J.B.O.: Machine learning methods in chemoinformatics. WIREs Comput. Mol. Sci. **4**, 468 (2014)
2. Martínez M.J., Ponzoni I, Díaz M.F., Vázquez G.E., Soto A.J.: Visual Analytics in Cheminformatics: User-Supervised Descriptor Selection for QSAR Methods. Journal of Cheminformatics **7**(39) (2015)

3. Le, T., Chandana Epa, V., Burden, F.R., Winkler, D.A.: Quantitative Structure-Property Relationship Modeling of Diverse Materials Properties. Chemical Reviews **112**(5), 2889 (2012)
4. Dorronsoro, I., Chana, A., Abasolo, M.A., Castro, A., Gil, C., Stud, M., Martinez, A.: CODES/Neural Network Model: a Useful Tool for in Silico Prediction of Oral Absorption and Blood-Brain Barrier Permeability of Structurally Diverse Drugs. QSAR Comb. Sci. **23**, 89 (2004)
5. Guerra, A., Páez, J.A., Campillo, N.E.: Artificial Neural Networks in ADMET Modeling: Prediction of Blood – Brain Barrier Permeation. QSAR Comb. Sci. **27**, 586 (2008)
6. DRAGON, Version 5.5, Talete srl, Milan, Italy (2007)
7. Soto, A.J., Cecchini, R.J., Vazquez, G.E., Ponzoni, I.: Multi-Objective Feature Selection in QSAR/ QSPR using a Machine Learning Approach. QSAR Comb. Sci. **28**, 1509 (2009)
8. Bengio, Y.: Learning Deep Architectures for AI. Foundations and Trends in Machine Learning **2**, 1 (2009)
9. Palomba,D., Cravero, F., Vazquez, G.E., Diaz, M.F.: Prediction of tensile strength at break for linear polymers applied to new materials development. In: Proceeding of the International Congress of Metallurgy and Materials - Sam-Conamet, Santa Fe, Argentina (2014)
10. Cravero, F., Vazquez, G.E., Diaz, M.F., Ponzoni I.: Modelado QSPR de propiedades mecánicas de materiales poliméricos empleando técnicas de reducción de variables basadas en algoritmos de aprendizaje automático (in Spanish). CAIQ. In: Proceeding of the Conference of Chemical Engineering. Buenos Aires, Argentina (2015)
11. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA Data Mining Software: An Update. SIGKDD Explorations **11**, 1 (2009)

Virtual Screening: A Challenge for Deep Learning

Javier Pérez-Sianes, Horacio Pérez-Sánchez and Fernando Díaz

Abstract The automated compound testing is currently the *de facto* standard method for drug screening, but it has not brought the great increase in the number of new drugs that was expected. Computer-aided compounds search, known as Virtual Screening, has shown the benefits to this field as a complement or even alternative to the robotic drug discovery. This paper will attempt to provide a comprehensive and structured survey that collects the most important proposals made so far along with what we think will be a key player in the future, the recent developments carried out in the deep learning field.

Keywords Drug discovery · Virtual Screening · Structure-based · Ligand-based · Machine Learning · Deep Learning

1 Introduction

Prior to the biomedical research boost driven by technological advances, structural bioinformatics tools already allowed to simulate molecules interactions based on

J. Pérez-Sianes
Departamento de Informática, University of Valladolid, Valladolid, Spain
e-mail: jpsianes@gmail.com

H. Pérez-Sánchez
Bioinformatics and High Performance Computing Research Group (BIO-HPC),
Computer Engineering Department, Universidad Católica San Antonio de Murcia (UCAM),
Guadalupe, Spain
e-mail: hperez@ucam.edu

F. Díaz(✉)
Departamento de Informática, Escuela de Ingeniería Informática,
University of Valladolid, Segovia, Spain
e-mail: fdiaz@infor.uva.es

M.S. Mohamad et al. (eds.), *10th International Conference on PACBB*,
Advances in Intelligent Systems and Computing 477,
DOI: 10.1007/978-3-319-40126-3_2

properties of quantum mechanics. These techniques became part of the procedures known as drug design and drug discovery, where this information is used to identify, design and optimize new drugs of pharmacological interest.

Drug search and discovery is a process which aims to find a molecule able to bind and activate or inhibit a molecular target; this target is usually a protein. Compounds that exceed a certain threshold in their capacity to strongly bind to a protein are called **lead compounds**. Traditionally this search was a manual process. Despite the criticism, currently the high-throughput screening (HTS) is the *de facto* standard method used in the search for lead compounds. By using robots, this technology allows researchers to test thousands of molecules, however, it is extremely expensive and it requires to own an extensive library of drugs and compounds. All this issues led to the emergence of computer tools to assist drug design.

The computation of interactions between molecules allows to reduce the number of compounds that have to be tested *in vitro*. The term Virtual Screening (VS) was coined to name this sort of *in silico* search, by analogy with the HTS. In order to do this kind of simulations, both the three-dimensional structure of the target receptor and the testing compound have to be known. Such methods are called Structure-based Virtual Screening (SBVS) [1]. If the structure of the molecular target is not available, a Ligand-based Virtual Screening (LBVS) strategy [2] can be used by searching similar molecules to compounds with known activity.

Machine learning (ML) is another important resource for drug discovery [3], it can be found mainly as a LBVS screening approach. These techniques require less computational resources than the calculation of molecule interactions and find more diverse hits than other similarity methods due to its generalization ability. AI and thereby machine learning methods are currently experiencing a strong resurgence. Deep learning (DL) and new approaches in neural networks have managed to make a quantitative and qualitative leap in this field of computer science [4]. In spite of this, these techniques have not achieved a great penetration in the areas of bioinformatics and computational biology.

This paper is intended to present to the reader the principal frameworks which encompass most of the proposals that can be found in the literature in a structured manner. Also, a brief introduction on the new artificial intelligence approaches is included in order to point out what we think can be a remarkable trend on the future of virtual screening.

2 Virtual Screening Background

Computational drug search can act as a complement or as an alternative to HTS testing and reduce costs by identifying false negatives. It has been observed that many of the VS strategies are more limited than the HTS for identifying sets of compounds with a great structural diversity, but sometimes they have found the best candidates on the hits of the simulations. There are various studies confirming the benefits of using computational methods in drug search.

Unlike the HTS approach, which is a sort of random search in a library of compounds, the VS is a knowledge-driven process. The most widespread classification of VS methods focus on this point, as shown in Table 1 (adapted from Bielska et al. [5]).

Table 1 Classification of VS methods based on the amount of information available (adapted from the one given by Bielska et al. [5])

	Known ligand(s)	*Unknown ligands*
Known structure of target or close homologue	structure-based VS: - protein-ligand docking - High Throughput Molecular Dynamics	*de novo* structure-based VS: - protein-ligand docking - High Throughput Molecular Dynamics
Unknown target structure	ligand-based VS: - few ligands: similarity search - several ligands: pharmacophore search, QSAR	virtual screening cannot be applied

The availability of structural information can aid to use specific non-exclusive strategies to address a virtual screening endeavor and there are two main types of strategies that encompass the most common screening methods: structure-based VS (which is in turn subdivided into *de novo* ligand methods) and ligand-based VS. In general, most authors classify their work in one of these three categories

1. Structure-based VS
2. Ligand-based VS
3. Combinatorial or structure-based *de novo* design

However, some of the procedures are important enough to be classified as a separate category, although they belong to one of these main classes. Therefore it is also common that some methods be cataloged in any of these other two categories:

4. Chemogenomics
5. Machine Learning

2.1 Structure-Based Virtual Screening

Modern sequencing technologies have facilitated the identification of numerous new targets. When we have their structural information, we can use it to perform a tentative computation of the interaction with other known molecules. Efforts in this kind of methods have succeed both in basic and applied research but still have several challenges ahead, as handling flexibility in the protein side.

Table 2 Structure-based virtual screening summary.

STRUCTURE-BASED VIRTUAL SCREENING		
ASSUMPTION	The knowledge of the molecular structure of the target drives the search of lead compounds.	
KEY NOTION	*Docking* as the fitting of two molecules in a favorable configuration (pose) to form a complex compound [6].	
Approach	Docking as a search and optimization task characterized by three facets:	
	Main issues	The *molecular representation* which defines the search space, e.g. atomic surfaces and grid representation.
		The *search methods* which define how to traverse the search space, e.g. exhaustive or blind search, randomized search methods, simulation methods.
		The *scoring function* which assesses the fitness of the found hits, e.g. force-field-based scores, empirical scoring functions, knowledge-based scoring function.

2.2 Ligand-Based Virtual Screening

If the 3D structure of the target is not available, the option is to apply a ligand-centered screening strategy, where the knowledge about active or non-active compounds will be used to retrieve other potentially active molecules based on similarity measures. This approach is broadly known as *similarity search*.

The key factor in this scheme is the definition of chemical similarity between molecules. The employed information must correlate with the molecule activity and its description should facilitate a quick comparison between compounds.

Recently there has been an increasing interest in the *combination of ligand-based and structure-based VS* methods to leverage all kinds of information available [8]. Basically, there are two schemes for hybridizing methods: the sequential scheme (a pipeline of methods) and the parallel scheme (a selection of best or coincident outcomes). One way or another, the developed tests yielded mixed results. The fact of applying different strategies together does not systematically improve the screening.

Table 3 Ligand-based virtual screening summary.

<table>
<tr><td colspan="4">LIGAND-BASED VIRTUAL SCREENING</td></tr>
<tr><td>ASSUMPTION</td><td colspan="3">Similarity-property principle: structurally similar molecules are expected to exhibit similar properties or activities [7].</td></tr>
<tr><td>KEY NOTION</td><td colspan="3">Chemical similarity between molecules.</td></tr>
<tr><td rowspan="8">Different approaches</td><td colspan="3">Similarity as a measure of the degree of pattern matching among compounds: direct alignment.</td></tr>
<tr><td colspan="3">Similarity as a measure of the statistical correlation among the features of each compound: QSAR techniques.</td></tr>
<tr><td rowspan="6">Main issues</td><td colspan="2">It can be viewed as an instance-based approach centered on establishing a regression or classification model of compounds.</td></tr>
<tr><td colspan="2">Different descriptors</td></tr>
<tr><td>1D descriptors</td><td>Scalar values quantifying compound features.</td></tr>
<tr><td rowspan="2">2D descriptors</td><td>Linear descriptors: binary fingerprints, real-valued vectors.</td></tr>
<tr><td>Tree-type descriptors.</td></tr>
<tr><td rowspan="2">3D descriptors</td><td>Interaction grids</td></tr>
<tr><td>Pharmacophores</td></tr>
</table>

2.3 Combinatorial or de novo Design

The technologies developed in the area of combinatorial chemistry allow to synthesize new compounds by combining various chemical elements and thus the building of large libraries of compounds to be screened, but the real breakthrough with regard to virtual screening has been the rational or *de novo* design [9].

Combinatorial libraries can be constructed following some logic to guide us in browsing through the vast search space of chemical compounds in order to find the best combination of molecular fragments which can be constituted in a whole ligand. Libraries constructed in this way usually provide a more appropriate source of test compounds, since they are more specific or they have greater structural diversity

Other ways to create *de novo* compounds have been pointed out, however, most methods fall into two basic categories: atom-based and fragment-based. Atom-based methods make up the molecule atom by atom while the fragment-based procedures use sets of predefined molecular building blocks which are connected

by a synthesis scheme. This second option is more engaging because it allows quickly build compounds in line with the design goal, but it tends to produce less specific designs.

2.4 Chemogenomics

Chemogenomics is more of a working methodology than another screening method. This VS approach suggests to work with compound classes and protein families as a whole instead of the classic one-ligand and one-protein view [10].

This fact introduces a novel paradigm which shifts from studying receptors as individual entities to a more global view. The structure of the proteins has a lot to do with their functions, so in the different groups or families there are receptors that exhibit multiple common features. Under the assumption that "similar receptors bind similar ligands" [11], given a target receptor of interest, drug discovery processes on chemogenomics perform screenings taking into account all compounds and ligands of similar receptors as well as the similar compounds to these ligands.

Just as in the similarity search methods discussed above, the chemogenomics framework need to establish similarity measures although in a more general manner. Usually the same measures than the one-target methods can be equally employed.

2.5 Machine Learning and Virtual Screening

Machine learning [12] is usually presented as a VS category but many of these algorithms are used in methods included in other categorizations by replacing their common estimations with intelligent models trained to exploit the available information. They generally need data on both active and non-active compounds, but this increased knowledge will help you get more accurate and diverse results.

The generation of QSAR models is one of the most popular applications of supervised ML in VS [13] since these algorithms exhibit a great potential for modeling complex nonlinear relationships. *Artificial neural networks* (ANN) have been one of the first ideas transferred to this area and neural nets such as the multilayer perceptron or probabilistic neural networks have been successfully applied in QSAR studies. In the nineties, *support vector machines* (SVM) appeared and they soon highlighted due to their generalization ability. Today is one of the main options to apply in the search for new active compounds and other stages of drug design.

Other relevant algorithms have also been moved to the cheminformatics area. The *decision trees* have received a special attention since generated models allow to interpret the results in terms of decision rules. They are mostly used as models for homogeneous ensembles because of their well-known sensitivity to changes in the data. Indeed, the *Random Forest* (RF) ensemble method has shown to be a competitive alternative and one of the main rivals of SVM. Other approaches like

the *k-Nearest Neighbors* and *Naive Bayes Classifiers* have been proven in supervised tasks like the establishment of QSAR relationships or the analysis of toxicity, although with less success.

Unsupervised techniques and optimization methods have been applied in a variety of tasks. *Self-organizing maps* (SOM) have shown their potential in toxicity studies and design of novel compounds. Search methods such as *genetic algorithms*, *ant colonies* and *particle swarm optimization* have been used for combinatorial design, building QSAR relationships or even conducting docking [14].

3 New Approaches to Artificial Intelligence

In recent years, new theories have emerged in the area of AI posing a new approach to classical machine learning and pattern recognition algorithms. This trend known as *deep learning* [4] has made news in mainstream and specialized media when this kind of algorithms have shown a clear mastery in various competitions and benchmarks and has caught the attention of leading big tech companies.

The fundamental basis of deep learning is rooted in another concept of machine learning, the relational learning. This concept refers to the transformation of the input data to some type of representation in order to obtain knowledge of such information to a higher level of abstraction [15]. From this idea, deep systems propose to separate learning in multiple layers; each one will learn an increasingly abstract and/or complex concept that is the starting point for learning in the next layer [16].

Currently the notion of "deepness" is closely tied to neural network architectures. Actually the deep networks have been around for many years but, except some prominent case, additional hidden layers appeared to offer no practical benefits and these ideas were soon discarded, first by the principle established by the *universal approximation theorem* [17], which states that a feed-forward network with a single hidden layer can approximate any multivariate continuous function, and secondly by two issues, above all, that appear when working with large networks:

1. *Vanishing gradients*: in gradient based learning algorithms, like backpropagation, as the information moves backwards in the net, gradients go becoming smaller.
2. *Overfitting*: when complex models are created is such way that they completely fit the training set, they will not make good generalizations on the predictions of new examples.

In 2006, Hinton [18] introduced a novel network architecture, called Deep Belief Networks (DBNs), which made feasible the training of multilayer neural networks. This breakthrough was shortly after accompanied by the proposals of Bengio's [19] and LeCun's [20] research groups. The main progress of these new models of neural networks has been the introduction of a pre-training phase using

an unsupervised learning algorithm that is applied greedily in a layer-wise fashion. By using certain unsupervised algorithms, it is achieved the encoding of the information in simpler and more compact forms of representation, thus reducing redundancies. These and other techniques like early stopping, weight penalties or dropout have helped overcome the aforementioned problems.

Despite the excellent results obtained in the Merck molecular activity challenge [21], a molecular activity prediction competition, it has not been done much research on the application of deep architectures in VS processes. Among the few papers published, there are two dedicated to the prediction of physicochemical and biological properties of drug-like compounds [22, 23] where both studies discuss the improvements in performance that these algorithms attain with respect to previous prediction models. For their part, some members of the winning team of the Merck challenge decided to continue this line of work and have presented two studies on QSAR modeling with deep neural networks (DNN) [24, 25]. In these studies, the use of combined multi-task training data sets is proposed to slightly outperform the current dominant RF. It is clear that the use of correlated data jointly can increase the power of learning models; these studies show that DNNs are capable to a better assimilation of possible shared high-level features than other ML models and use simultaneously different data that are not close enough in order to be combined into a single dataset. However, it is pointed out that although the standard DNNs parameter settings provide good results, its predictive ability is variable under different settings and cross-validation is not effective as a setup technique for configuring it under real conditions. In addition, it has also been found that the unsupervised pre-training, a key issue in deep learning, usually degrades DNNs results in QSAR tasks, so other methods have been used to avoid overfitting.

4 Conclusion

Computer-aided drug discovery is becoming increasingly important in the field of drug design as a means to reduce time and cost in the search for novel lead compounds. However, there is no single VS method that yields better results than others in any test and HTS remains as the fundamental technique of automated drug design.

SBVS is limited by issues like the impossibility of obtaining the 3D structure of certain molecules and the difficulty of including the flexibility of the protein in the models. LBVS techniques, on other hand, are highly dependent on the type of molecular representations and compound classes. The ML approach represents a trade-off between computational speed and structural diversity of the found hits and ML tools have become very popular because they are available, efficient and simple to interpret.

In recent years, other proposals have arisen to mitigate to some extent the shortcomings of these methods, e.g. the integration of SBVS and LBVS or the rational *de novo* design. Whatever strategy you choose, machine learning can

always contribute, and new approaches to AI, especially the relational and deep learning, have entered strongly into the field, generating great expectation about the possibilities that they may offer. Their greater power and ability to automatically learn complex concepts and high-level abstractions could overcome the limitations attributed to ML algorithms by being unable to access to the underlying laws of quantum mechanics [26]. Multi-task learning may be the gateway to a better understanding of the subjacent mechanisms of biomolecules activity and binding, and it could also surpass obstacles such as discontinuous QSAR [27] but, in general, there is still much to research on the use of DL in virtual screening processes and our further work will be centered on the development of deep learning-based methods for the virtual screening process.

Acknowledgements This work was partially supported by (i) the Fundación Séneca del Centro de Coordinación de la Investigación de la Región de Murcia under Project 18946/JLI/13 and by the Nils Coordinated Mobility under grant 012-ABEL-CM-2014A, in part financed by the European Regional Development Fund (ERDF) and (ii) the project Platform of integration of intelligent techniques for analysis of biomedical information (TIN2013-47153-C3-3-R) from the Spanish Ministry of Economy and Competitiveness.

References

1. Lionta, E., Spyrou, G., Vassilatis, D., Cournia, Z.: Structure-Based Virtual Screening for Drug Discovery: Principles, Applications and Recent Advances. Curr. Top. Med. Chem. **14**, 1923–1938 (2014)
2. Ripphausen, P., Nisius, B., Bajorath, J.: State-of-the-art in ligand-based virtual screening. Drug Discov. Today **16**, 372–376 (2011)
3. Gertrudes, J.C., Maltarollo, V.G., Silva, R.A., Oliveira, P.R., Honorio, K.M., da Silva, A.B.F.: Machine learning techniques and drug design. Curr. Med. Chem. **19**, 4289–4297 (2012)
4. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature **521**, 436–444 (2015)
5. Bielska, E., Lucas, X., Czerwoniec, A., Kasprzak, J.M., Kaminska, K.H., Bujnicki, J.M.: Virtual screening strategies in drug design - methods and applications. Biotechnologia **92**, 249–264 (2011)
6. Kitchen, D.B., Decornez, H., Furr, J.R., Bajorath, J.: Docking and scoring in virtual screening for drug discovery: methods and applications. Nat. Rev. Drug Discov. **3**, 935–949 (2004)
7. Johnson, A.M., Maggiora, G.M.: Concepts and Applications of Molecular Similarity. John Wiley & Sons Inc., New York (1990)
8. Drwal, M.N., Griffith, R.: Combination of ligand- and structure-based methods in virtual screening. Drug Discov. Today. Technol. **10**, e395–e401 (2013)
9. Schneider, G., Böhm, H.J.: Virtual screening and fast automated docking methods. Drug Discov. Today **7**, 64–70 (2002)
10. Kubinyi, H.: Chemogenomics in drug discovery. Ernst Schering Res. Found. Workshop, 1–19 (2006)

11. Klabunde, T.: Chemogenomic approaches to drug discovery: similar receptors bind similar ligands. Br. J. Pharmacol. **152**, 5–7 (2007)
12. Flach, P.: Machine Learning: The Art and Science of Algorithms that Make Sense of Data (2012)
13. Butkiewicz, M., Mueller, R., Selic, D., Dawson, E., Meiler, J.: Application of machine learning approaches on quantitative structure activity relationships. In: 2009 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology, pp. 255–262. IEEE (2009)
14. Melville, J.L., Burke, E.K., Hirst, J.D.: Machine learning in virtual screening. Comb. Chem. High Throughput Screen. **12**, 332–343 (2009)
15. Bengio, Y., Courville, A., Vincent, P.: Representation learning: a review and new perspectives. IEEE Trans. Pattern Anal. Mach. Intell. **35**, 1798–1828 (2013)
16. Bengio, Y.: Learning Deep Architectures for AI. Found. Trends Mach. Learn. **2**, 1–127 (2009)
17. Hornik, K., Stinchcombe, M., White, H.: Multilayer feedforward networks are universal approximators. Neural Networks **2**, 359–366 (1989)
18. Hinton, G.E., Osindero, S., Teh, Y.-W.: A fast learning algorithm for deep belief nets. Neural Comput. **18**, 1527–1554 (2006)
19. Vincent, P., Larochelle, H., Bengio, Y., Manzagol, P.-A.: Extracting and composing robust features with denoising autoencoders. In: Proceedings of the 25th International Conference on Machine Learning - ICML 2008, pp. 1096–1103. ACM Press, New York (2008)
20. Poultney, C., Chopra, S., Lecun, Y.: Efficient learning of sparse representations with an energy-based model. In: Advances in Neural Information Processing Systems (NIPS 2006), pp. 1137–1144. MIT Press (2006)
21. Kaggle - Merck Molecular Activity Challenge. https://www.kaggle.com/c/MerckActivity
22. Lusci, A., Pollastri, G., Baldi, P.: Deep Architectures and Deep Learning in Chemoinformatics: The Prediction of Aqueous Solubility for Drug-Like Molecules. J. Chem. Inf. Model. **53**, 1563–1575 (2013)
23. Xu, Y., Dai, Z., Chen, F., Gao, S., Pei, J., Lai, L.: Deep Learning for Drug-Induced Liver Injury. J. Chem. Inf. Model. (2015). 151013124508007
24. Dahl, G.E., Jaitly, N., Salakhutdinov, R.: Multi-task Neural Networks for QSAR Predictions. CoRR. abs/1406.1 (2014)
25. Ma, J., Sheridan, R.P., Liaw, A., Dahl, G.E., Svetnik, V.: Deep neural nets as a method for quantitative structure-activity relationships. J. Chem. Inf. Model. **55**, 263–274 (2015)
26. Ramakrishnan, R., von Lilienfeld, O.A.: Machine Learning, Quantum Mechanics, and Chemical Compound Space (2015). http://arxiv.org/abs/1510.07512
27. Maggiora, G.M.: On outliers and activity cliffs–why QSAR often disappoints. J. Chem. Inf. Model. **46**, 1535 (2006)

A Primary Study on Application of Artificial Neural Network in Classification of Pediatric Fracture Healing Time of the Lower Limb

Sorayya Malek, R. Gunalan, S.Y. Kedija, C.F. Lau, Mogeeb A.A. Mosleh, Pozi Milow, H. Amber and A. Saw

Abstract In this study we examined the lower limb fracture in children and classified the healing time using supervised and unsupervised artificial neural network (ANN). Radiographs of long bones from 2009 to 2011 of lower limb fractures involving the femur, tibia and fibula from children ages 0 to 13 years, with ages recorded from the date and time of initial injury was obtained from the pediatric orthopedic unit in University Malaya Medical Centre. ANNs was developed using the following input: type of fracture, angulation of the fracture, displacement of the fracture, contact area of the fracture and age. Fracture healing time was classified into two classes that is less than 12 weeks which represent normal healing time in lower limb fractures and more than 12 weeks which could indicate a delayed union. This research was designed to evaluate the classification accuracy of two ANN methods (SOM, and MLP) on pediatric fracture healing. Standard feedforward, back-propagation neural network with three layers was used in this study. The less sensitive variables were eliminated using the backward elimination method, and the ANN network was retrained again with minimum variables. Accuracy rate, area under the curve (AUC), and root mean square errors (RMSE) are the main criteria used to evaluate the ANN model results. We found that the best ANN model results was obtained when all input variables were used with overall accuracy percentage of 80%, with RMSE value of 0.34, and AUC value of 0.8. We concluded here that the ANN model in this study can be used to classify

S. Malek(✉) · S.Y. Kedija · C.F. Lau · M.A.A. Mosleh · P. Milow
Intitute of Biological Science, Faculty of Science, University of Malaya, Kuala Lumpur, Malaysia
e-mail: {sorayya,pozimilow}@um.edu.my, kedija@siswa.um.edu.my, francesco_lau@hotmail.com, mogeebmosleh@yahoo.com

R. Gunalan · H. Amber · A. Saw
Faculty of Medicine, University of Malaya, Kuala Lumpur, Malaysia
e-mail: roshan@um.edu.my, dramber84@gmail.com, sawaik@hotmail.com

M.S. Mohamad et al. (eds.), *10th International Conference on PACBB*,
Advances in Intelligent Systems and Computing 477,
DOI: 10.1007/978-3-319-40126-3_3

pediatric fracture healing time, however extra efforts are required to adapt the ANN model well by using its full potential features to improve the ANN performance especially in the pediatric orthopedic application.

Keywords Artificial Neural Networks · Lower limb fractures · Immature skeleton · Kohonen self organizing maps

1 Introduction

A fracture is defined in the break in the continuity of the bone, usually involving the cortex of the bone. Fractures in children (aged 0 to 13 years) have considerably different features as opposed to fractures in adults. Skeletal trauma accounts for 15% of all injuries in children [1]. In children, an abnormal healing time may signal a non-accidental injury or an underlying medical condition affecting bone healing [2]. While rates have been published for a normal bone healing process in adults, there has been less literature with regards to pediatric fractures. Correlating healing time with the chronologic history of injury, may aid in injuries that are healing abnormally or may indicate non-accidental injury [3-5].

Applications of Artificial Neural Network (ANNs) have been reported in orthopedic field [6, 7, 8, 9]. SOM an unsupervised ANN has been used to classify the dataset for osteoporosis classification problem of high and low osteoporosis risk [9]. Self-organizing feature map (SOM) proof its efficiency in knowledge discovery and exploratory data analysis process for the unsupervised learning. The SOM has also showed it is suitability as an excellent tool in the visualization of high dimensional data [15]. SOM can be used to reduce the dimensions of data of a high level of complexity and plots the data similarities through clustering technique [16]. However ANNs applications have not been widely applied in the pediatric orthopedic field. We hypothesized in this study that the ability of estimating pediatric fracture healing time of the lower limb could be improved by using ANNs. Lower limb bones can be divided into four sections; the femur, tibia, fibula and the bones of the foot. Long bone fractures are described with reference to the direction of the fracture line in relation to the shaft of the bone. [10]. Few articles reported the evaluation of classification on pediatric fracture healing on the basis of radiographic fracture and statistical approach to determine healing rates. Pediatric bone physiology indicates that children healing time rate is faster compared to adults [10]. Correlating healing time with the chronologic history of injury, may aid abnormal healing or even suspected non-accidental injury. The aim of this study is to investigate the feasibility of application of ANNs to predict the classified healing time of pediatric fracture of lower limbs since injury to unite and become fully healed.

2 Materials and Methods

2.1 Data

A collection of four years of patient data and radiographs from the years 2009, 2010, 2011 and 2014 respectively were obtained from the University Malaya Medical Centre, Orthopaedic Department in Kuala Lumpur, Malaysia. Radiographs of fractured bones (femur, tibia and fibula) from 57 samples of children of ages less than 13 years were included, with ages recorded from the time of initial injury. Any individuals demonstrating comorbidity or any systemic disorder, which may affect the bone healing rate, were excluded from the study.

Parameters used in the study are lateral (sagittal plane) and anterior (coronal plane) angle, displacement and contact area and age. The measurement of a bone fracture can be done in three possible ways. They are (1) angulation; (2) displacement and (3) contact area. The measurement is taken based on radiograph features [11]. Angulation describes the direction of the distal bone and degree of angulation in relation to the proximal bone. Displacement of fracture is defined in terms of the abnormal position of the distal fracture fragment in relation to the proximal bone. Contact area is described as how much the bone is in contact with each other. Angulation, displacement and contact area are interrelated to each other [12].

Days of healing (calculated as the number of days between fracture occurrence and radiograph evidence of healing). The fracture healing time was categorized based on the notion that all types of lower limb fractures in children will heal within 12 weeks. The characteristics of fracture healing time in this study were then classified into two classes, normal healing time, and delayed union. Where normal healing time for all lower limb fracture is taken less than 12 weeks regularly, and delayed union is taken more than 12 weeks [10]. The option to consider the exact healing time compared to a period of time is more accurate but not practical in a clinical setting. This is because patients are followed up at intervals to assess the progression of their injuries. Therefore, the exact time of healing will fall between clinic appointments in which radiograph images are taken. Data statistics summary statistics used in this study is presented in table 1.0.

Table 1 Summary statistic of variable used for ANN model development

	Min	Max	Median	Mean	StD
Anterior angle (degree)	0.00	1.85	0.00	0.4528	0.56
anterior displacement (mm)	0.00	21.00	5.90	5.926	5.89
anterior contact area (mm)	0.00	2.00	1.820	1.548	0.69
lateral angle (degree)	0.00	50.00	1.00	6.25	9.54
lateral displacement (mm)	0.00	22.00	2.80	5.18	6.31
lateral contact area (mm)	0.00	100.0	64.00	58.4	38.99
Age (year)	0.11	13	6	6.51	4.28

2.2 ANN Model Development

Both supervised and unsupervised ANN was applied in this study using Neural Network Toolbox in MATLAB [13]. Back-propagation learning with one hidden layer was employed as supervised ANN part. ANN model is constructed with three main layers, where input layer included 7 nodes, hidden layers included 10 nodes, and output layers included 1 node. The network was trained with scaled conjugate gradient [14].

Three data sets are used for the ANN model development in this study a training set, a test set, and a validation set [15]. Backward elimination method was used to eliminate insignificant variable. Eliminations process involves starting with all candidate variables, testing the deletion of each variable using a chosen model comparison criterion, deleting the variable (if any) that improves the model the most by being deleted, and repeating this process until no further improvement is possible. The inputs are ranked using Pearson correlation coefficient before carrying out backward elimination process. Root mean square error (RMSE), receiving operator characteristic (ROC), area under the curve (AUC) and accuracy rate were the main assessment criteria adopted to evaluate this study results.

Self-Organizing Maps (SOM) is used in this study [16] to ordinate fracture input variables with respect to healing time. As a result of the training of the unsupervised ANN, the Euclidian distance between the inputs are calculated and visualized as distance matrix (U-matrix). SOM reduces data dimensions by producing a map of 1 or 2 dimensions which plot the similarities of the data by grouping similar data items together. Thus, SOM reduce dimensions and display similarities. This enables the discovery or identification of features or patterns of most relevance through data reduction and projection. The u-matrix representation of SOM visualizes the distances between neurons. The distance between the adjacent neurons is calculated and presented with different colorings between the adjacent nodes [17]. Light areas can be though as clusters and dark areas as cluster separators. One can directly find the cluster in the input data using this indication.

3 Results and Discussion

Figure 1.0 below illustrates the Self Organizing Maps to view relationship of variables used in this study. The final quantization and topographic error are 0.333 and 0.035. It is a measure of how good the map can fit the input data and how well topology of the data is preserved. The appropriate map is expected to yield the smallest average quantization error. The U-matrix in Figure 1.0 illustrates fracture data used in this study is separated in two clusters light areas can be though as clusters and dark areas as cluster separators. This conforms to the classification used in this study to classify the healing time into two classes. Each component plane shows the values of a single vector component in all nodes of the map, hence, together they can be seen as constituting the whole U-matrix. It can been seen that variables that illustrates correlation based on the similarities same pattern

with healing time in weeks ; 1) anterior angle and lateral angle and opposite correlation with healing week are, 2) anterior contact area and lateral contact area. There is no distinctive pattern between healing weeks with lateral displacement, anterior displacement and age. However there is a pattern between age and displacement. It can be seen that older children (8-10 year old) heals faster provide when the anterior and lateral angle of the fracture is low regardless of the displacement and the contact area.

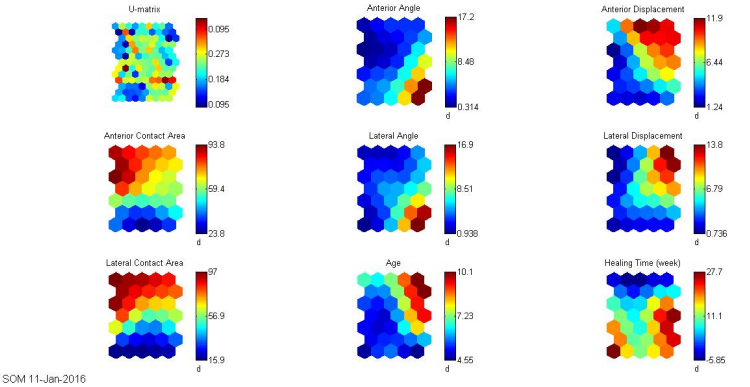


Fig. 1 : U-matrix and component planes of pediatric fracture data

Supervised ANN model using all seven input parameters that are anterior and lateral angle, displacement, contact area, and age reported the most accurate results. There was no further improvement in results by using backward elimination method. The results for percentage of accuracy among training, and testing sets are reported as 88.9 % and 80.9% respectively. The RMSE reported for training, and testing are 0.3386, and 0.3575 respectively. Fig. 2 illustrates confusion matrix and ROC curve for the classification of fracture healing time for the suitable model by using all input variables. Class one in Fig. 2 refers to fracture healing time less than 12 weeks and class 2 in Fig. 2 refers to fracture healing time more than 12 weeks. The overall accuracy is 80.7%, where the study model was classified correctly 46 samples over 57. Meanwhile the misclassification rate was 19.3%, where 11 samples only were misclassified out of 57. About 7 samples from class one are classified as class two and 4 samples from class two are being classified as class one. The testing data accuracy is reported as 88.9% with 11.1% misclassification. All the samples in class one are classified correctly. Class two reported 85.7% accuracy this is because one sample from class two is classified as class one. The overall AUC for class one is 0.74 and class 2 is 0.75.

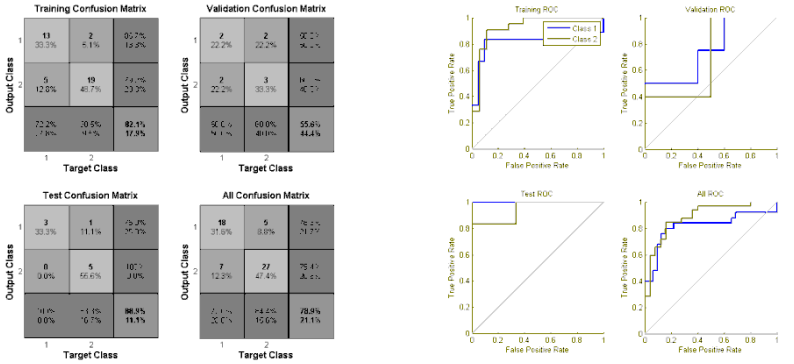


Fig. 2 Confusion Matrix and ROC graph using all input variables

4 Discussion

In this study we have shown that clinical fracture data can be visualized in a 2-dimensional representation using SOM. This allows the clinician to place a new patient within the context of similar cases. Results conform to clinical data summary that older children (8-10 year old) heal slightly faster when the anterior and lateral angle of the fracture is low, regardless of the displacement and the contact area [18]. Therefore smaller angulation in the coronal and sagittal plane play important role in the time to union of the long bone. In lower limb fractures, if the plane of motion is large it will delay the osteogenic process of bone healing. Displacement and contact area of the fracture, do not affect the healing time significantly. This is because of the thick osteogenic periosteal layer present in immature skeleton which facilitates bone healing and is not compromised with the displacement of the fracture [19].

This can use as a tool for straightforward diagnostic reasoning in conjunction with presentation with lower limb fracture. It may be a useful screening mechanism for detecting children with risk with obstructed longer healing time which may require special care. At this stage, it is not possible to claim the results here as universal application. Since the study is based upon limited clinical data which only takes into account the urban population around the city of Kuala Lumpur. The pattern of fractures however, is comparable to other population sets and if used within a validation system and continually recreated as more data is collected.

In the present study for the supervised ANN, 7 variables are identified that are significant to predict fracture healing time are anterior and lateral angle, displacement, contact area, and age. In the literature, it has reported that these variables play a role in the healing times of fractures in the immature skeleton. A fracture heals in response to biological principles and mechanical principles. The biological aspect of a child is generally the same with the corresponding age group. However, the pattern of the fractures may have an impact on optimal healing time. In general, in lower limb fractures, the alignment of the fracture along the

mechanical axis is the aim of management of these fractures. A change in angulation (either in coronal or sagittal plane), displacement of the fracture sites, or the contact area of the fracture are usually modified to the lowest possible value in order for optimum time to bone union [19]. ANN constructed using all input variables in this study provided higher results and further reducing the number of variables using backward elimination did not improve the results. The results achieved in this study is compared to studies to predict hip fractures in elderly population using ANN [20] which reported accuracy rate of almost 80%, with and AUC of 0.8 almost similar to the results achieved in this study. The justification of using MLP in this study on limited number of datasets available is based on Milan et. al [21] on his work on experimental analysis in the limited training data set conditions. In this study he justified the use of the MLP only for classification of the simple and well-defined dataset that would avoid the over fitting and generalization problem. The data in this study is considered as simple well defined not complex and furthermore leave on out method was used to compare the results and the results obtained are almost similar. The results obtained in this study could be further improved as the present study has some limitations. First, type of fracture and sex of the patient were not included in the predictive models. This exclusion might lower the performance of the classifiers. Second, some clinical risk factors were not included, such as any underlying medical condition, the condition of the soft tissue surrounding the fracture and the method of treatment used. This includes if reduction was performed prior to immobilization of the fracture. The time to union was also categorized somewhat broadly, as the patients are seen in the clinic at various stages of bone healing and not at the exact time the bone achieved union. Besides, this sample size was limited about 57 cases as not many lower limb pediatric cases are reported. The variation in ages of the patient within the study was also similar, negating the fact that age of the patient plays a role in the time to bone union. Future enhancement of the ANN design using different algorithm such as support vector machine topology or using different method could improve the accuracy rate with considerable amount of data.

5 Conclusions

The lower limb fracture healing time could be assessed by using supervised and unsupervised ANN analyses. With adequate model construction ANN may produce better results. However, application of ANN seems have not been adapted to its full potential especially in pediatric fracture healing. More studies are required to further improve the model in this study in terms of performance, accuracy, and other machine learning methods are required to be validated externally such as support vector machine.

Acknowledgment This study was funded by University of Malaya grant RG370-15AFR. Dataset cannot be downloaded due to doctor-patient confidentiality, in which the patients have consented for their data to be used, but not for their personal information or demographics to be made public.

References

1. Staheli, L.: Fundamentals of Pediatric Orthopedics, 3rd edn. Lippincott Williams and Wilkins, Pennsylvania (2003)
2. University of Rochester Medical Center, May 27, 2015. http://www.urmc.rochester.edu/encyclopedia/content.aspx?ContentTypeID=90&ContentID=P02760
3. Patton, D.F.: Fractures and orthopaedics. Churchill Livingstone, Edinburgh (1992)
4. Hobbs, C.J.: Fractures. Br. Med. J. **298**, 1015–1018 (1989)
5. Kempe, R.S., Silverman, F.N., Steele, B.F., Droegemueller, W., Silver, H.K.: The battered child syndrome. Am. J. Med. Sci. **181**(1), 17–24 (1962)
6. Shi, L., Wang, X.C., Wang, Y.S.: Artificial neural network models for predicting 1-year mortality in elderly patients with intertrochanteric fractures in China. Brazilian Journal of Medical and Biological Research **46**, 993–999 (2013). doi:10.1590/1414-431X20132948
7. Kukar, M., Kononenko, I., Silvester, T.: Machine learning in prognosis of the femoral neck fracture recovery. Artif. Intell. Med. **8**(5), 431–451 (1996)
8. Taylor, R.J., Taylor, A.D., Smyth, J.V.: Using an artificial neural network to predict healing times and risk factors for venous leg ulcers. J. Wound Care **11**(3), 101–105 (2002)
9. Sharpe, P.K., Caleb-Solly, P.: Self organising maps for the investigation of clinical data: A case study. Neural Computing and Applications **7**(1), 65–70 (1998). ISSN 0941-0643, http://eprints.uwe.ac.uk/19201
10. Ogden, J.A.: Injury to the immature skeleton. In: Touloukian, R. (ed.) Pediatric Trauma, 2nd edn. John Wiley & Sons, New York (1990)
11. Ogden, J.A.: Skeletal Injury in the Child, 2nd edn. WB Saunders, Philadelphia (1990)
12. Staheli, L.: Practice of Pediatric Orthopedics, 2nd edn. Lippincott Williams & Wilkins, Philadelphia (2006)
13. Natick, MA, R20013, MathWorks
14. Meiller, M.F.: A scaled conjugate gradient algorithm for fast supervised learning. Neural Netw. **6**, 525–533 (1993)
15. Smith, A.E., Mason, A.K.: Cost estimation predictive modeling: Regression versus neural network. The Engineering Economist **42**(2), 137–161 (1997)
16. Kohonen, T.: Self- Organization and Associative Memory, 3rd edn. Springer-Verlag (1989)
17. Hollmén, J.: Process modeling using the selforganizing map, Master's thesis, Department of Computer Science, Helsinki University of Technology (1996)
18. McKibbin, B.: The biology of fracture healing in long bone. J. Bone Joint Surg. **60B**, 150–162 (1978)
19. Ryöppy, S.: Injuries of the growing skeleton. Ann. Chir. Gynaecol. Fenn. **61**, 3–10 (1972)
20. Tseng, W.-J., Hung, L.-W., Shieh, J.-S., Abbod, M.F., Lin, J.: Hip fracture risk assessment: artificial neural network outperforms conditional logistic regression in an age- and sex-matched case control study. BMC Musculoskeletal Disorders **14**, 207 (2013)
21. Milan, Z.M., Markovic, M.M., Adreja, B.S.: A performance analysis of the multilayer perceptron in limited training data set conditions. IEEE (1997)

Visual Exploratory Assessment of Class C GPCR Extracellular Domains Discrimination Capabilities

Martha I. Cárdenas, Alfredo Vellido and Jesús Giraldo

Abstract G protein-coupled receptors (GPCRs) are integral membrane-bound proteins. They are divided in five main classes with Class C members receiving recently special attention because of their involvement in many neurologic diseases. These receptors are composed of the seven-helix transmembrane domain (typical of all GPCRs) and a large extracellular domain where the endogenous ligand binds. In the present absence of crystal structures for complete Class C receptors, their primary sequences can provide limited but useful information that can be used for subtype discrimination in first instance. In this paper, we show that the extracellular part of these sequences provides as effective a discrimination as the complete sequence. With that purpose, we describe a process of exploratory sequence visualization using different data transformations and manifold learning techniques for dimensionality reduction. Class discriminability is assessed using an entropy-based measure.

Keywords GPCRs · N-terminus · Data visualization · GTM · Kernel-GTM

1 Introduction

G protein-coupled receptors (GPCRs) are integral membrane-bound proteins responsible for signal transduction from outside to inside the cell. This mediator's role makes

M.I. Cárdenas(✉) · A. Vellido
Departament de Ciències de la Computació, Universitat Politècnica de Catalunya,
08034 Barcelona, Spain
e-mail: mcardenas@cs.upc.edu

J. Giraldo
Institut de Neurociències and Unitat de Bioestadística, Universitat Autònoma de Barcelona,
Cerdanyola del Vallès, 08193 Barcelona, Spain

A. Vellido
CIBER-BBN, Cerdanyola del Vallès, Barcelona, Spain

M.I. Cárdenas—This research was partially funded by MINECO TIN2012-31377, ERA-NET NEURON PCIN-2013-018-C03-02 and SAF2014-58396-R research projects.

M.S. Mohamad et al. (eds.), *10th International Conference on PACBB*,
Advances in Intelligent Systems and Computing 477,
DOI: 10.1007/978-3-319-40126-3_4

these proteins to be involved in many physiological and pathological conditions. As a consequence, GPCRs are one of the main targets of pharmaceutical research with about 30 percent of current drug targets belonging to this protein superfamily [1]. GPCRs are classified into 5 families or classes: Class A, Class B (Secretin), Class C (Glutamate), Adhesion and Frizzled [2].

In the present study we focus on Class C GPCRs. These receptors are currently being given particular attention because they are involved in many neurologic disorders [3]. The mechanistic understanding of the functional behaviour of a protein commonly depends on the accurate knowledge of its crystal 3D structure. This knowledge is limited in the case of membrane proteins such as GPCRs, and only in recent years, some GPCR structures have been solved, mostly those from Class A [4]. Class C GPCRs are particularly complex structures. Currently, only the 7TM domains of two Class C GPCRs [5, 6] and several extracellular domains have been determined, but separately. In the present absence of the crystal structure for a complete Class C receptor, their primary structure (that is, the amino acid sequence) can provide limited but useful information, which in a first instance can be used for subtype discrimination. Fortunately, this information is publicly available from several curated databases. All GPCRs are characterized by sharing a common seven-helix transmembrane (7TM) domain, which is responsible for G protein binding and activation. In addition, Class C GPCRs bear a large domain in the extracellular part of the receptor (N-terminus) called the Venus Flytrap (VFT) and, in most of their subfamilies, a cystein rich domain (CRD) connecting both.

Class C GPCRs have a rich taxonomy of subfamilies. The automatic discrimination and classification of their subfamilies becomes a problem on its own right [7] for which machine learning techniques can provide an informed solution. Knowledge extraction from Class C GPCRs primary structure information can be approached through exploratory data visualization, which can be informative in domains in which data structure is not fully known or uncertain, leading to hypothesis generation via inductive reasoning [8]. A first problem obviously arises for visualization: the transformation of varying-length sequential symbolic data into formats that are suitable for multivariate data analysis. Those transformations might use the complete sequences in unaligned form or apply methods of multiple sequence alignment (MSA). Both will be used in our experiments. These transformations lead to a second problem: that of the high dimensionality of the transformed data, making direct data visualization impossible. In this scenario, dimensionality reduction (DR) methods are necessary; out of the many DR families of techniques available to the analyst, and following previous research on this problem [9], our work focuses here on manifold learning methods.

The hypothesis in this study is that the results of Class C GPCR subfamily discrimination should differ depending on whether we use the complete primary sequence or, instead, we use only the extracellular N-terminus. Related to this, we also hypothesize that the N-terminus should be almost as good as the complete sequence in terms of subfamily discrimination. The reason for this lies on VFT including the site where endogenous ligands for Class C GPCRs bind and, as a consequence, a diversity in

AA sequence is expected. A secondary hypothesis is that these differences should intuitively be observed through manifold learning-based visualization.

2 Materials and Methods

2.1 Materials

Class C of GPCRs is subdivided into seven main subfamilies, namely: Metabotropic Glutamate (MGlu) receptors, Calcium sensing (CS), GABA-B, Vomeronasal (VN), Pheromone (Ph), Odorant (Od) and Taste (Ta).

The specific data set investigated in this study was extracted from the GPCR-DB [10] database system for GPCRs which divides the GPCR superfamily into five major families (A to E) based on the ligand types, functions, and sequence similarities. It includes 1,510 GPCR sequences that belong to Class C, from which 1,252 include an extracellular N-terminal domain description. Their distribution of cases by subfamily is summarized in Table 1.

Table 1 Number of available sequences (those including N-Terminus) in each of GPCR Class C subfamilies.

Class C Type	Subfamily name	Cases	Class C Type	Subfamily name	Cases
Type 1	Metabotropic Glutamate	282	Type 5	Pheromone	333
Type 2	Calcium Sensing	45	Type 6	Odorant	80
Type 3	GABA-B	156	Type 7	Taste	63
Type 4	Vomeronasal	293			

Primary sequences have to be transformed for their subsequent visualization analysis using DR methods. Two transformations were considered here; the first one uses the whole-length, unaligned sequences and is called amino acid composition (AAC) transformation [11]. It consists on calculating the frequencies of the 20 amino acids of the sequence *alphabet*. As such, it ignores the sequential information itself (i.e., the relative position of the amino acids). Despite this, its use has previously yielded surprisingly solid results [11, 12]. The second transformation is a common MSAand, as such, it uses only partial information from each sequence.

2.2 Methods

As described in the introduction, our investigation focuses on the exploratory visualization of GPCRs from their primary sequences, using nonlinear manifold learning. Specifically, we used Generative Topographic Mapping (GTM, [13]), a probabilistic

generative model. In this paper, we take advantage of its capability to adapt to different data types and use it in two variants for the two different data transformations described in previous sections: the AAC and the MSA.

The standard GTM models multivariate data by enveloping them in a low-dimensional manifold (2-D for data visualization). Such manifold is expressed as a discrete network of cluster centroids, or data prototypes, that can also be seen as centres of distributions (isotropic Gaussians in the standard definition). Thus, GTM can also be described as a manifold-constrained mixture of distributions. The model is expressed as a nonlinear mapping from a low-dimensional latent visualization space (2-D) into the observed data space, in the form $y = \Phi(u) W$, where y is a vector in a D-dimensional data space, Φ is a set of M basis functions, u is a point in the visualization space and W is a matrix of adaptive weights w_{md}. As previously stated, the model is probabilistically described; the probability distribution for data point x in $X = \{x_1, ..., x_N\}$ with $x \in \Re^D$, generated by a latent point u, is defined as an isotropic Gaussian noise distribution with a common inverse variance β:

$$p(x|u, W, \beta) = \left(\frac{\beta}{2\pi}\right)^{D/2} \exp\left\{-\frac{\beta}{2} \|x - y(u, W)\|^2\right\} \tag{1}$$

Integrating out the latent variables u, we can obtain $p(x)$ and the corresponding likelihood of the model. Standard maximum likelihood methods can be used for parameter estimation. Details for the standard GTM can be found in [13]. As part of the parameter estimation process, the probability of each of the K latent points u_k for the generation of each data point x_n can be quantitatively calculated as the *responsibility* r_{kn}:

$$r_{kn} = \mathrm{P}(k|x_n, W, \beta) = \frac{\exp\left\{-\frac{\beta}{2} \|x_n - y_k\|^2\right\}}{\sum_{k'=1}^{K} \exp\left\{-\frac{\beta}{2} \|x_n - y_{k'}\|^2\right\}} \tag{2}$$

The calculation of r_{kn} (which indicates that, in our case, each GPCR sequence n has a probability of being mapped to the area of the visualization space represented by latent point k) allows us to visualize our data in different ways, including a mode projection expressed as x_n: $k_n^{mode} = \arg\max_{\{k_n\}} r_{kn}$. This standard GTM model is used in our experiments to model and visualize the AAC-transformed unaligned sequences.

The kernel-GTM (KGTM) [14] is a kernelized version of GTM that is specifically well-suited to the analysis of symbolic sequences such as those characterizing proteins. This is achieved describing sequence similarity through a kernel function specifically designed for the job, as described in [14]. KGTM is used here to model and visualize the MSA-transformed sequences.

3 Results

Our experiments are organized according to two different dimensions. First, we analyzed the available sequences according to two different transformations using two different methods: unaligned sequences are transformed according to the AAC method and analyzed using the standard GTM, while KGTM is used to analyze sequences transformed by MSA. Second, we use two approaches to assess the results: exploratory visualization for a qualitative interpretation of the global (sub)structure of subfamilies, complemented by a quantitative assessment of the level of subfamily discrimination, based on an entropy measure.

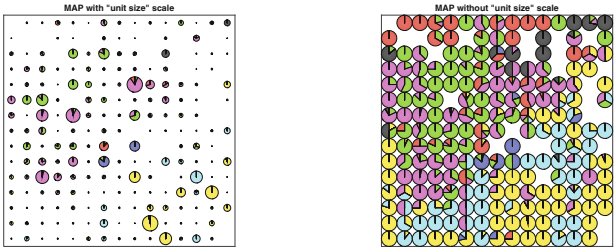
The mode projection of the AAC-transformed data on the standard GTM visualization map is shown in Fig.1(a) for the complete Class C GPCR sequences and in Fig.1(b) for the extra-cellular N-terminus of the same sequences. Correspondingly, the mode projection of the MSA-transformed data on the KGTM visualization map is shown in Fig.1(c) for the complete Class C GPCR sequences and in Fig.1(d) for the extra-cellular N-terminus of the same sequences. In order to visually assess the level of subfamily mixing in each of (K)GTM latent points, modes are represented as pie charts. These mode projections are, in the end, a simplified representation in which each sequence is mapped to the latent point of highest responsibility r_{kn}. It is also interesting to use the richer probabilistic information provided by the model to inspect the *responsibility maps* of individual sequences through visualization of the distribution of r_{kn} values on the (K)GTM maps. Some examples are shown in Fig. 2.

3.1 Entropy

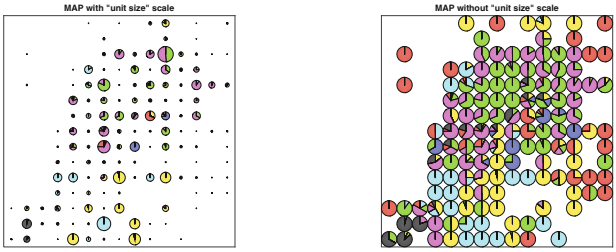
In order to complement the qualitative exploratory visualization of the Class C GPCR sequences, we describe here a quantitative assessment of subfamily overlapping, based on an entropy-based measure that is suitable for discrete clustering visualizations such as those provided by the GTM variants. In our experiments, it should be understood as a measure of Class C heterogeneity. When (K)GTM map areas are completely subfamily-specific (that is, when no two sequences of different subfamilies are assigned to the same (K)GTM latent point), the corresponding entropy will be zero, whereas high entropies will characterize highly overlapping subfamilies.

Generally speaking, entropy depends on the probability that the model attributes to the source. In the case of (K)GTM, the total entropy for a given latent point k in the visualization space will be expressed as $S_k = -\sum_{j=1}^{C} p_{kj} ln p_{kj}$, where j is one of the seven GPCR Class C subfamilies and $p_{kj} = \frac{m_{kj}}{m_k}$, where m_k is the number of sequences in cluster k and m_{kj} is the total number of sequences in cluster k which belong to subfamily j.

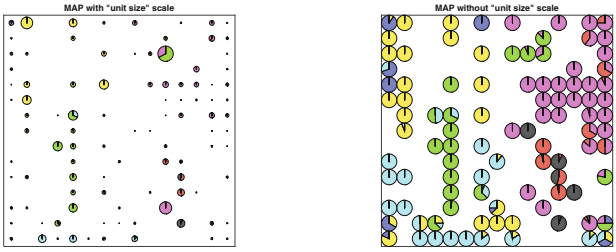
Then, the total entropy for a subfamily j for all units in the map can similarly be defined as $E_j = -\sum_{k/m_{kj}>0} p_{kj} ln p_{kj}$ and, finally, the Weighted-Average Entropy



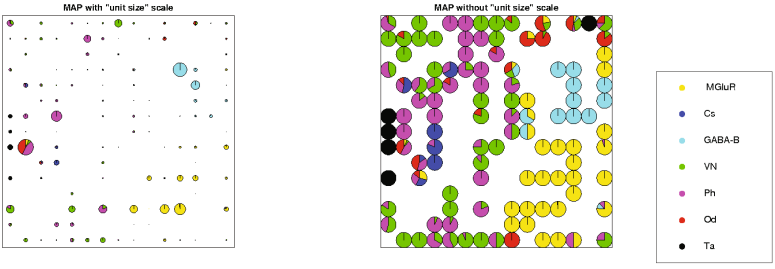
(a) GTM visalization map for AAC-transformed complete sequences



(b) GTM visualization map for AAC-transformed N-terminals

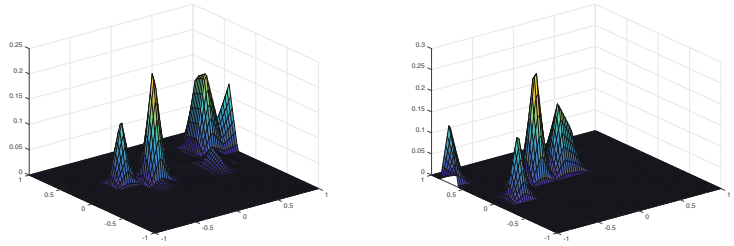


(c) KGTM visalization map for MSA-transformed complete sequences



(d) KGTM visualization map for MSA-transformed N-terminals (left and centre).

Fig. 1 Visualization maps of the different data *mode projections*. The left and right columns display the same data representation; their difference is that, in the maps on the left, the size of the pie chart encodes the ratio of sequences assigned to a given latent point, therefore providing visual clues about the spatial distribution of relative data density. Subfamily labels for all maps are shown in the bottom-right legend.



(a) *XP002942445*, N-terminus sub-sequence (b) *XP002940939*, complete sequence

Fig. 2 Visualization of r_{kn} for some example AAC-transformed sequences (standard database names included) from subfamilies VN (*a*) and Ph (*b*).

Table 2 Subfamily entropies for N-terminal Domain and Complete GPCR.

	GPCR Complete								N-terminal Domain							
	MGlu	**CS**	**GB**	**VN**	**Ph**	**Od**	**Ta**	E_{wa}	**MGlu**	**CS**	**GB**	**VN**	**Ph**	**Od**	**Ta**	E_{wa}
AAC	3.65	0	2.75	15.05	15.90	4.36	0.43	0.37	5.91	0.35	3.40	9.52	12.23	4.20	2.38	0.46
MSA	3.43	1.89	3.56	2.34	3.63	2.99	0.91	0.18	2.98	1.42	1.42	8.10	9.07	4.7	0	0.26

E_{wa} as $E_{wa} = \sum_k S_k \frac{m_k}{N}$, where N is the total number of sequences in the analyzed dataset. Table 2 summarizes the entropies per subfamily E_j and the E_{wa} for each of the transformed datasets in our study.

4 Discussion

The mode projections for all datasets in Fig. 1 reveal some striking differences. Overall, the GTM representation of the AAC-transformed sequence projections is far more distributed than that of the KGTM of the MSA-transformed sequence projections, with many latent points taking responsibility for only a few sequences. Interestingly, and specially for the GTM, the N-terminus projection is much more compact than that of the complete sequence, involving far fewer latent points. In all cases, a limited number of latent points concentrates a relatively large number of sequences; this is particularly the case in the KGTM MSA-transformed representation.

The examples of individual r_{kn} in Fig. 2 correspond to cases from different subfamilies in which the probability of assignment of sequences to latent points is clearly multi-modal. This illustrate the way the models handle uncertainty. Multimodal cases with lower maxima are most frequent in sub-families with high levels of overlapping, such as VN, Ph and Od.

The entropy results reported in Table 2 only partially corroborate our starting hypothesis. The overall entropy of the KGTM representation of the MSA-transformed sequences is lower than the corresponding entropy of the GTM representation of the AAC transformation, both for the complete sequences and for the N-terminus. In all cases, the complete sequences yield lower entropies than the N-terminus; this means that the N-terminus only partially retains the subfamily discrimination capabilities of the complete sequence. The inspection of the entropies per subfamily reveals a less clear-cut picture. For the GTM with AAC, the use of the N-terminus increases the entropy for the easier-to-discriminate subfamilies (mGlu, CS, GB, Ta), while it decreases the entropy for the most overlapping ones (VN, Ph, Od). For the KGTM with MSA is precisely the other way around: the use of the N-terminus decreases the entropy for the easier-to-discriminate subfamilies and increases the entropy for the most overlapping ones. In any case, MSA keeps the entropies of the overlapping subfamilies at rather low values. All in all, this apparently contradictory behaviour should be investigated in more detail in future research.

5 Conclusions

GPCR cell membrane proteins of Class C have created great expectations in pharmacology as targets of drug design. They have a heterogeneous sub-family structure and, because of the absence of crystal structures including all the domains of any of these receptors, the investigation on their primary sequential structures can be of great help. The discrimination of these subfamilies has been shown to have clear limits due to overlapping. In this study, we have investigated subfamily separability through visual exploration using manifold learning methods and an entropy-based measure. The adequacy of the separate use of the extra-cellular N-terminus domain for subfamily discrimination purposes has been assessed. Results are mixed and somehow inconclusive, showing that overall discriminability decreases when only the N-terminus is used, but with mixed patterns depending on the data transformation and manifold learning model.

References

1. Overington, J.P., et al.: How many drug targets are there? Nature Reviews Drug Discovery **5**, 993–996 (2006)
2. Alexander, S.P., et al.: The Concise Guide to PHARMACOLOGY 2015/16: G protein-coupled receptors. British Journal of Pharmacology **172**, 5744–5869 (2015)
3. Kniazeff, J., et al.: Dimers and beyond: The functional puzzles of class C GPCRs. Pharmacology & Therapeutics **130**(1), 9–25 (2011)
4. Cooke, R.M., et al.: Structures of G protein-coupled receptors reveal new opportunities for drug discovery. Drug Discovery Today **20**(11), 1355–1364 (2015)
5. Wu, H., et al.: Structure of a class C GPCR metabotropic glutamate receptor 1 bound to an allosteric modulator. Science **344**(6179), 58–64 (2014)

6. Doré, A.S., et al.: Structure of class C GPCR metabotropic glutamate receptor 5 transmembrane domain. Nature **551**, 557–562 (2014)
7. Gao, Q.B., Ye, X.F., He, J.: Classifying G-protein-coupled receptors to the finest subtype level. Biochemical and Biophysical Research Communications **439**(2), 303–308 (2013)
8. Vellido, A., et al.: Seeing is believing: the importance of visualization in real-world machine learning applications. In: Proceedings of ESANN 2011, pp. 219–226 (2011)
9. Cárdenas, M.I., et al.: Visual characterization of misclassified class C GPCRs through manifold-based machine learning methods. Genomics and Computational Biology **1**(1), e19 (2015)
10. Horn, F., et al.: GPCRDB: an information system for G protein-coupled receptors. Nucleic Acids Research **26**, 275–279 (1998)
11. Sandberg, M., et al.: New chemical descriptors relevant for the design of biologically active peptides. A multivariate characterization of 87 amino acids. Journal of Medicinal Chemistry **41**, 2481–2491 (1998)
12. Cárdenas, M.I., Vellido, A., Giraldo, J.: Visual interpretation of class C GPCR subtype overlapping from the nonlinear mapping of transformed primary sequences. In: Proceedings of IEEE BHI 2014, pp. 764–767 (2014)
13. Bishop, C.M., Svensén, M., Williams, C.K.I.: GTM: The Generative Topographic Mapping. Neural Computation **10**, 215–234 (1998)
14. Olier, I., Vellido, A., Giraldo, J.: Kernel Generative topographic mapping. In: Proceedings of ESANN 2010, pp. 481–486 (2010)

Development of a Machine Learning Framework for Biomedical Text Mining

Ruben Rodrigues, Hugo Costa and Miguel Rocha

Abstract Biomedical text mining (BTM) aims to create methods for searching and structuring knowledge extracted from biomedical literature. Named entity recognition (NER), a BTM task, seeks to identify mentions to biological entities in texts. Dictionaries, regular expressions, natural language processing and machine learning (ML) algorithms are used in this task. Over the last years, @Note2, an open-source software framework, which includes user-friendly interfaces for important tasks in BTM, has been developed, but it did not include ML-based methods. In this work, the development of a framework, BioTML, including a number of ML-based approaches for NER is proposed, to fill the gap between @Note2 and state-of-the-art ML approaches. BioTML was integrated in @Note2 as a novel plug-in, where *Hidden Markov Models*, *Conditional Random Fields* and *Support Vector Machines* were implemented to address NER tasks, working with a set of over 60 feature types used to train ML models. The implementation was supported in open-source software, such as *MALLET*, *LibSVM*, *ClearNLP* or *OpenNLP*. Several manually annotated corpora were used in the validation of BioTML. The results are promising, while there is room for improvement.

Keywords Biomedical text mining · Named entity recognition · Machine learning

1 Introduction

Nowadays, the life sciences produce large amounts of information spread in scientific literature and databases. The biomedical literature contains non-structured

R. Rodrigues(✉) · M.Rocha
Centre of Biological Engineering, University of Minho, Braga, Portugal
e-mail: pg25227@alunos.uminho.pt

R. Rodrigues · H. Costa
Silicolife, Lda, Braga, Portugal

M.S. Mohamad et al. (eds.), *10th International Conference on PACBB*,
Advances in Intelligent Systems and Computing 477,
DOI: 10.1007/978-3-319-40126-3_5

data [1], written in natural language, making the extraction of high-quality information a difficult challenge. Indeed, biomedical researchers spend large amounts of time extracting useful information from literature. The Biomedical Text Mining (BTM) field is concerned with the extraction of high-quality information from literature from the biological and biomedical domains. BTM emerged to create tools and methodologies that can automate and reduce time-consuming tasks when searching for information lying in biomedical literature [2].

Information Extraction (IE) has the ability to extract high-quality information from text streams. Named Entity Recognition (NER), a task in IE, aims to identify bio-entities in text streams [3]. NER tasks can be performed by distinct approaches, like lexicon-based, rule-based or machine learning-based techniques. NER performance can be tested and validated against gold standard corpora for specific case studies containing curated annotations [4].

Several NER methods have been developed with different advantages and limitations [3]. ML techniques have proven to be reliable, fast, scalable and automated processes. These can be used to perform NER tasks over biomedical literature [5, 7], requiring curated training sets to learn models. Hidden Markov Models (HMM) [8], Conditional Random Fields (CRF) [9] and Support Vector Machines (SVM) [6] are common ML models used in BioTM tasks [5, 10].

@Note2 [11] is a multi-platform BTM workbench written in Java, which encompasses the most important Information Retrieval and Information Extraction tasks. However, ML-based methodologies were not duly exploited in the available versions, a limitation that will be addressed by this work.

Indeed, the main goal of this work is to construct a framework that integrates ML algorithms addressing BTM tasks to fill the gap between @Note2 operations and state-of-the-art ML approaches. To make that possible, this study aims to:

- Create a ML framework with the capacity to train different models from annotated corpora and applying them to raw text for NER purposes;
- Create tools to evaluate and compare the performance of different models for the annotation of bio-entities, enabling the comparison of ML based approaches with other methods already implemented;
- Create a plug-in for @Note2 which allows the connection of ML tools with the remaining @Note2 structures, also defining appropriate user interfaces for the ML operations integrated within @Note2s architecture;
- Validate the overall framework with gold standard corpora.

2 Methods

Machine Learning (ML) is a sub-field of computer science and statistics that has been applied to solve problems in different scientific areas, being deeply related to Artificial Intelligence and Optimization. ML addresses the creation of mathematical models from data [1]. There are several advantages regarding the use of ML methods,

such as the possibility to retrieve accurate results in an automated, fast and scalable way. However, ML methods have some limitations due to the dependency on the input data to train the models and issues as overfitting or underfitting [12]. Metrics as precision, recall and F-scores can be used to evaluate the performance of ML methods. These are calculated regarding a confusion matrix resulting from the application of an ML model to a set of examples, encompassing true and false positives and negatives [12].

Regarding BTM, ML methodologies require a training process using annotated data to create a model that can then be used to classify/find the terms in raw (unmarked) text. Commonly used models include Hidden Markov Models (HMM) [8], Conditional Random Fields (CRF) [9] and Support Vector Machines (SVM) [13]. These are typically trained with manually curated corpora. The major limitation of these techniques is the fact that creating curated corpora is a time-consuming process and a trained model may only be applied to a specific problem (i.e. is difficult to generalize to other biological contexts).

ML algorithms can be split in classifiers and transducers. Classifiers are used to classify text tokens as entities of interest (or not) based in the features that characterize the token and its neighborhood. Transducers are also used to annotate the token, but the classification is done not only based in the token features but also in a sequence of previous tokens' features. SVMs are classifiers which calculate, from training data, a hyperplane that separates the examples in distinct classes with a maximal margin separation, building a linear space through the use of a kernel function. On the other hand, HMMs and CRFs are possible transducers. An HMM is a statistical Markov model in which probability distributions are used to model data from time series observations [8]. The training of a HMM consists in model parameters adjustment, from the available features. The hidden states represent the possible NER classes and the observations are the features. CRFs are undirected graphical models used to segment and label sequence data [9]. This model presents advantages over HMMs since the strong independence assumptions made in HMMs can be relaxed in CRFs [14]. The CRF training has similarities with the one from HMMs, but the CRF supports more features in the same model. CRFs can be used for NER tasks, in which the labels for the states are features related to the entity annotation [7, 15].

The ML features can be classified in several groups according to the token characterization that is performed. Examples of those feature groups are orthographic, semantic, morphological, and sentence structure features, among others.

3 Implementation

The main purpose of BioTML is to perform NER tasks using ML approaches. Its main pipelines allow (i) the creation of ML models from annotated corpora and (ii) the capability to predict new annotations in unannotated documents. The conceptual structure of the framework was devised to accomplish these two pipelines (training

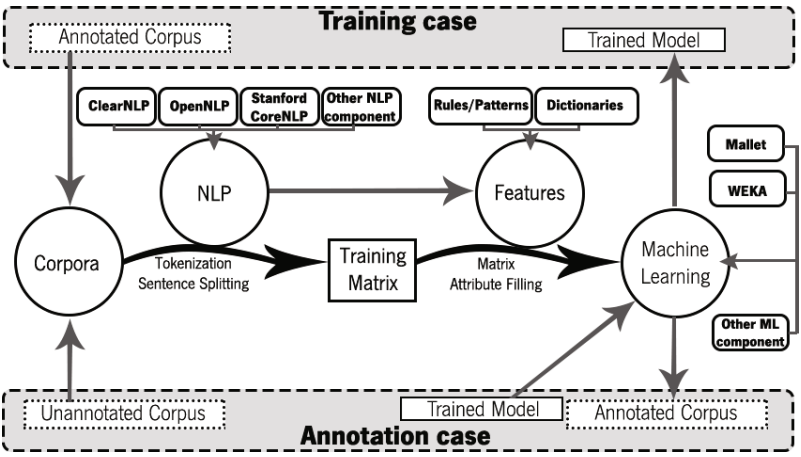


Fig. 1 Conceptual structure of the BioTML framework. The two BioTML main pipelines are the training and annotation processes represented in gray. Regarding the selected pipeline, the BioTML system uses a corpus (annotated or not), processes the tokenization using one of the implemented NLP systems and generates a matrix of features that is used in the ML module to train the model. All modules can be extended with new NLP systems, features or ML systems.

and prediction/ annotation), being divided in 4 main modules: Corpora, NLP, features and machine learning (shown in Figure 1).

The corpora module includes the corpus, document, sentence, token and annotation data structures, being responsible for storing documents and annotations. The NLP module includes components provided by open-source software (like OpenNLP [16], ClearNLP [17] and Stanford CoreNLP [18]). The use of this module is addressed to process tokenization of text streams in the corpora module, to create features and to generate a prediction matrix during annotation in the ML module. The features module includes dictionaries, patterns/rules, and NLP components that allow the creation of new features (e.g. part of speech, lemmas, chunks, dependency parsing, etc.). The features are created for each sentence and token from the corpus in model training. The machine learning module includes algorithms to train models provided by Java-based software packages (like MALLET [19]), evaluation components to test the models and annotators to apply the models over corpora.

This conceptual structure can be summarized in the two main pipelines: one that takes an annotated and curated corpus to create a model for NER, and the other that takes a model and an unannotated corpus to perform an NER task and returns an annotated corpus. Those pipelines were integrated on @Note2 in the form of a novel plug-in that allows the connection of both platforms. In the training operation, an annotated corpus is retrieved from @Note2 platform and converted into a BioTML corpus. This data structure is used as input in BioTML to train a model. In the annotation operation, the @Note2 API is used to retrieve a @Note2 unannotated corpus that is converted into a BioTML corpus. BioTML receives this unannotated corpus as input, the model linkage, settings and configurations to create an annotated

corpus. The results can then be viewed using the functionalities provided by @Note2. Figure 2 shows some screenshots showing the main operations of this plug-in. The plug-in is available by installing the latest @Note2 version on http://www.anote-project.org/, the installation steps and plug-in documentation are described in http://darwin.di.uminho.pt/anote2/wiki/index.php/Machine_Learning_biotml.

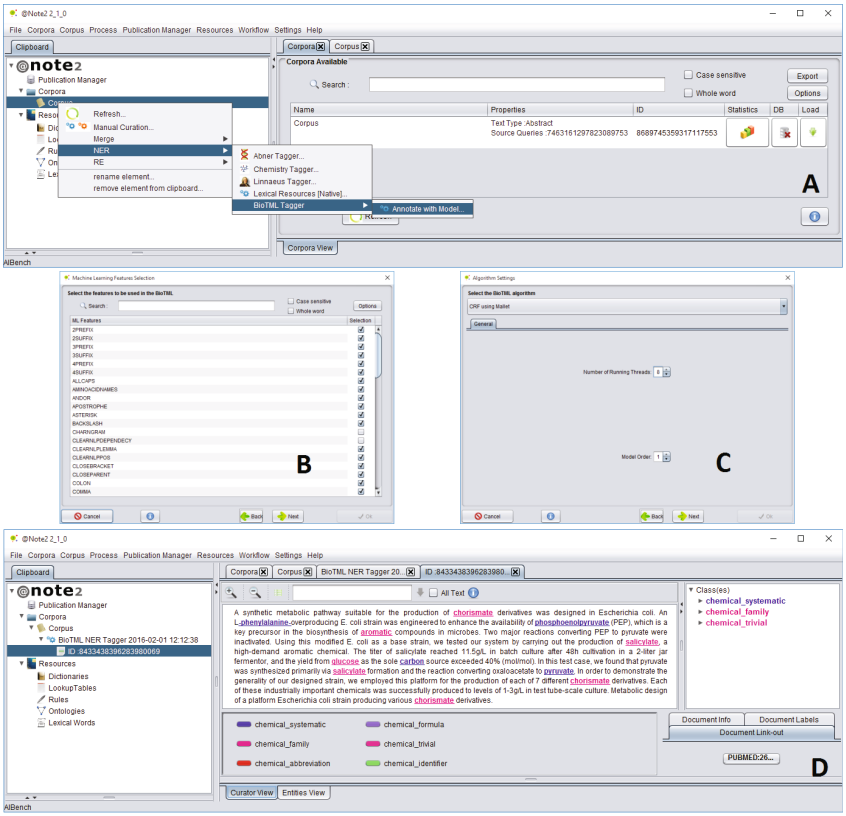


Fig. 2 Screenshots of BioTML plug-in. A - The plug-in GUI is accessed using a corpus instance from @Note2 clipboard. B - The feature types used for model training can be selected. C - The algorithm used for model creation is defined (HMM, CRF or SVM). D - A document annotated using a previously trained BioTML model.

4 Case Studies

4.1 BioTML Validation Using JNLPBA Corpus

BioTML was firstly validated using the JNLPBA corpus of the BioNLP/ NLPBA 2004 Shared Task [20]. The prediction capabilities of BioTML were tested using the training set to create the NER model and the test set to predict the annotations, using

the provided evaluation tool. The full corpus contains 5 types of NER annotations: protein, DNA, RNA, cell line and cell type. For each class, an NER model was trained using all features. The achieved prediction scores are given in Figure 3 and a comparison of BioTML against other systems is provided, namely Gimli [15] and the best system in the challenge [21].

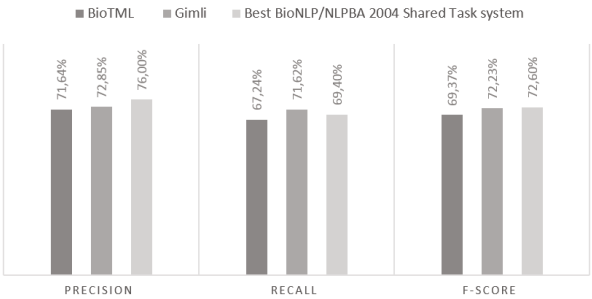


Fig. 3 Overall evaluation scores compared against other annotation systems.

As observed, the system has F-scores approximately 3% lower than both contenders. Although BioTML was not capable to perform at the level of the best systems, the results are promising since the difference is not large and the systems used in the comparison are among the best available. Indeed, being the first results of a novel platform, there is much room for improvements, since the features can be optimized for each class and no post-processing was performed.

4.2 BioCreative V: BioTML Used for Annotation of Chemical Entities in Patents

BioCreative proposes community challenges to evaluate BTM systems. Our framework was submitted to BioCreative V CHEMDNER-patents task [22], being used to perform the detection of chemical mentions in patents. To evaluate BioTML, we applied an approach in which CRFs trained with a set of specific features obtained by a feature optimization process (defined for each chemical class) were used to train ML models. We only used CRFs for this case study because the training of SVM algorithms was slower compared to the CRFs training and the HMM algorithms performed lower scores in the internal prediction results.

Each participant could submit 5 test set annotations. For each run, our framework predicted annotations using models trained over different sets of curated annotations. The predictions produced by each model were submitted and evaluated. The results including the systems' scores, BioTML test set scores, BioTML best prediction scores in the development set (trained using only the training set) and baseline scores

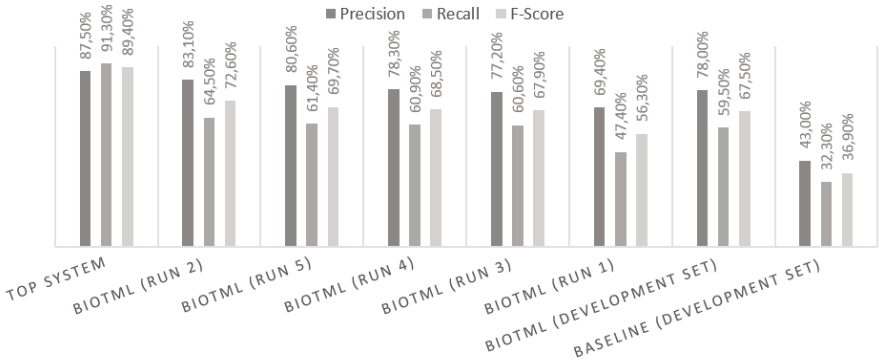


Fig. 4 BioCreative V task CHEMDNER-patents results and development results comparison against baseline results.

are given in Figure 4. The baseline scores were created by a case sensitive lookup list of all chemicals present in the training data. With these scores, we could verify if the training and development data share the same entity names and if our system could perform prediction of new entities present in the development set and not in the training set.

The results show that BioTML performed with high precision but low recall. A possible reason can be overfitting and a solution could be the use of other features addressed to identify a larger range of chemicals (e.g. word clustering). Another problem was the absence of a post-processing step, since a few annotation errors have a large impact in the precision, and could be corrected by post-processing steps as odd parenthesis verification in systematic chemical entities, re-annotation of chemical names using the predicted chemical names and dictionaries, addition of rules to only allow annotations that are between spaces or a selected punctuation. Overall, the system is not only capable to perform chemical annotation predictions with high precision, but also performs predictions with better recall than systems that only use dictionaries/lookup lists.

5 Conclusions and Further Work

ML approaches have been developed and implemented for different areas of BTM, including NER, using HMMs, CRFs and SVMs, among other algorithms. In this work, a ML-based framework dedicated to NER tasks, named as BioTML, has been developed and validated. The main innovation of this work is the creation of a modular framework, integrating several NLP and ML systems that can be enhanced with further systems using the provided API. Also, it is capable to predict NER annotations

with high performance. The BioTML integration into @Note2 allows using ML approaches for NER in an user-friendly environment.

ML limitations like the necessity of representative and large datasets, high variety of feature types, optimization of selected features and fine-tuning of the settings are important and were taken into account in BioTML design. Although the achieved results are promising, the system could be improved with the implementation of more features, post-processing steps (e.g. with dictionary matching and/or regular expression rules) and other NER approaches which could bring more options to create models fitted to the given datasets.

Additionally, the framework can be improved in several points, which will be addressed in the future, including the implementation of further ML algorithms, of a command line interface or of automatic feature selection algorithms. Also, relation extraction tasks can be handled by BioTML without major changes.

Acknowledgments This work is co-funded by the North Portugal Regional Operational Programme, under the "Portugal 2020", through the European Regional Development Fund (ERDF), within project SISBI- Ref NORTE-01-0247-FEDER-003381.

References

1. Feldman, R., Sanger, J.: The Text Mining Hand Book - Advanced Approaches in Analysing Unstructured Data (2007)
2. Shatkay, H., Craven, M.: Mining the biomedical literature (2012)
3. Nadeau, D., Sekine, S.: A survey of named entity recognition and classification. Lingvisticae Investigationes **1–20**, 2007 (1991)
4. Kim, J.D., Ohta, T., Tateisi, Y., Tsujii, J.: GENIA corpus - A semantically annotated corpus for bio-textmining. Bioinformatics **19** (2003)
5. Eom, J., Zhang, B.: PubMiner : Machine Learning-based Text Mining for Biomedical Information Analysis. Genomics **2**, 99–106 (2004)
6. Takeuchi, K., Collier, N.: Bio-medical entity extraction using support vector machines. Artificial Intelligence in Medicine **33**, 125–137 (2005)
7. Bundschus, M., Dejori, M., Stetter, M., Tresp, V., Kriegel, H.P.: Extraction of semantic biomedical relations from text using conditional random fields. BMC Bioinformatics **9**, 207 (2008)
8. Ramage, D.: Hidden Markov models fundamentals. Standford CS229 Section Notes, pp. 1–13 (2007)
9. Sutton, C.: An Introduction to Conditional Random Fields. Foundations and Trends in Machine Learning **4**(4), 267–373 (2012)
10. Torii, M., Wagholikar, K., Liu, H.: Detecting concept mentions in biomedical text using hidden Markov model: multiple concept types at once or one at a time? Journal of Biomedical Semantics **5**, 3 (2014)
11. Lourenço, A., Carreira, R., Carneiro, S., Maia, P., Glez-Peña, D., Fdez-Riverola, F., Ferreira, E.C., Rocha, I., Rocha, M.: @Note: A workbench for Biomedical Text Mining. Journal of Biomedical Informatics **42**(4), 710–720 (2009)
12. Batanlar, Y., Özuysal, M.: Introduction to machine learning. Methods in Molecular Biology **1107**, 105–128 (2014)
13. Quan, C., Wang, M., Ren, F.: An unsupervised text mining method for relation extraction from biomedical literature. PLoS ONE **9**(7), 1–8 (2014)
14. Pereira, F., Lafferty, J., Mccallum, A.: Conditional random fields: probabilistic models for segmenting and labeling sequence data. In: Proceedings of 18th International Conference on Machine Learning, (ICML), pp. 282–289 (2001)

15. Campos, D., Matos, S., Oliveira, J.L.: Gimli: open source and high-performance biomedical name recognition. BMC Bioinformatics **14**, 54 (2013)
16. Morton, T., Kottmann, J., Baldridge, J.: OpenNLP: A Java-based NLP Toolkit (2005)
17. Choi, J.D.: Optimization of Natural Language Processing Components for Robustness and Scalability. PhD thesis, University of Colorado at Boulder (2012)
18. Manning, C.D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S.J., McClosky, D: The stanford coreNLP natural language processing toolkit. In: Proceedings of 52nd Annual Meet. Assoc. Comput. Linguistics: System Demonstrations, pp. 55–60 (2014)
19. McCallum, A.K.: MALLET: A Machine Learning for Language Toolkit (2002)
20. Kim, J.D., Ohta, T., Tsuruoka, Y., Tateisi, Y., Collier, N.: Introduction to the bio-entity recognition task at JNLPBA. In: Proceedings of Intern. Joint Workshop Natural Language Processing in Biomedicine and Its Applications, pp. 70–75 (2004)
21. Zhou, G., Su, J.: Exploring deep knowledge resources in biomedical name recognition. In: Workshop on Natural Language Processing in Biomedicine and Its Applications at COLING, pp. 96–99 (2004)
22. Krallinger, M., et al.: Overview of the CHEMDNER patents task. In: Proceedings of the Fifth BioCreative Challenge Evaluation Workshop, pp. 63–75 (2015)

Sequence Retriever for Known, Discovered, and User-Specified Molecular Fragments

S. Sagar and J. Sidorova

Abstract Typically, biological and chemical data are sequential, for example, as in genomic sequences or as in diverse chemical formats, such as InChI or SMILES. That poses a major problem for computational analysis, since the majority of the methods for data mining and prediction were developed to work on feature vectors. To address this challenge, a functionality of a *Statistical Adapter* has been proposed recently. It automatically converts parsable sequential input into feature vectors. During the conversion, insights are gained into the problem via finding regions of interest in the sequence and the level of abstraction for their representation, and the feature vectors are filled with the counts of interesting sequence fragments, – finally, making it possible to benefit from powerful vector-based methods. For this submission, the *Sequence Retriever* has been added to the Adapter. While the Adapter performs the conversion: sequence → vector with the counts of interesting molecular fragments, the Retriever performs the mapping: molecular fragment → sequences from the database that contain this fragment.

Keywords Sequence retrieval · Parsing · Bioactivity

1 Introduction

Databases containing sequential data are increasing in both size and complexity, and there is anticipation for a big bulk of scientific knowledge to be discovered from the collected data. The problem is that the majority of powerful methods for data-mining and prediction, e.g. [1], were developed for vectors[1], not graphs or

S. Sagar · J. Sidorova(✉)
Department of Computer Science and Engineering,
Blekinge Institute of Technology, Karlskrona, Sweden
julia.a.sidorova@gmail.com

[1] A vector is a one-dimensional array of fixed size to represent samples in a data set, for example: *0,1,10,56,0,1,toxic.*

M.S. Mohamad et al. (eds.), *10th International Conference on PACBB*,
Advances in Intelligent Systems and Computing 477,
DOI: 10.1007/978-3-319-40126-3_6

sequences of variable length, and these methods are not directly applicable, when it comes to processing sequential data, as in genome processing or *in silico* prediction of chemical activity. In the literature, sequential input has been processed in different ways depending on the modeling assumptions and problem settings.

- Firstly, syntactic pattern recognition for context-free[2] strings has been tried [9], which is learning a formal language per recognition class and then mapping new samples to the class to which the distance is smallest. Syntactic pattern recognition can be used if a grammar can be observed in a natural way [12]. Forcing modeling on data, e.g. imposing linear ordering, hampers the performance [13].
- The most widely used syntactic method is Hidden Markov Models (HMMs) [2]. For example, HMMs have been used to discover chromatin states [3] and protein regions with distinct biological functions [4]. The limitation is that modeling with HMMs treats sequences as one-dimensional strings of independent, uncorrelated symbols. Although this approach is computationally convenient, many biological phenomena have more complex structure than regular.
- In order to precisely emphasize the need to take into account the interactions between different parts of a sequence, kernel methods are used. Kernels work as an adapter with a black-box inner machinery. We take the metaphor of an *adapter*, that is, a device that enables normally incompatible devices to work together. A good kernel maps sequences (or graphs) with a small edit-distance to closer values. The drawback is that kernel methods are hard to interpret, although some recent efforts have been undertaken to add interpretability, e.g. [5].
- Fourthly, an adapter with transparent and fixed inner machinery provides an explicit sequence (or graph) mapping into the feature vectors, which are filled with the counts of the fragments from a closed list known to be critical from a biological or chemical perspective. Defining such new fragments (or "features") in specific research domains is an active research field, e.g. for chemical activity [6,7].

- Recently, a transparent adapter with *flexible* machinery was proposed [10], in which the fragments to be counted do not come from a closed list, which had been previously compiled by a scientist, but such fragments are automatically segmented from sequential data. The method works in the following steps: 1) acquisition of a formal grammar per recognition class; 2) comparison of the grammars in order to find fragments of interest represented as sequences of terminal and/or non-terminal symbols and filling the feature vector with their counts; and 3) hierarchical feature selection and hierarchical classification, deducing and accounting for the domain taxonomy. That is, the method segments interesting sequences from databases. This submission extends this work with a mechanism called Sequence Retriever to search for the

[2] The Chomsky hierarchy of structural complexity of sequences: regular $\subseteq$ context-free (a palindrome, a sequence of balanced parentheses) $\subseteq$ context-sensitive (a natural language) $\subseteq$ recursive $\subseteq$ recursively enumerable.

compounds that contain the found fragments from other databases. This task cannot be achieved via a substring search or the search with regular expressions through SMILES sequences, because the fragments of interest can contain several nonadjacent subsequences of known length.

The Statistical Adapter and Sequence Retriever have a Linux command line interface and freely available code in Java (on request from the corresponding author). The rest of the paper is organized as follows. In Section 2 a sequential representation of chemicals is explained to serve as the parsable sequence input. The Statistical Adapter and Retriever are explained in Section 3 and Section 4, respectively. Finally, the discussion is held and the conclusions are drawn in Section 5.

2 Sequential Representation of Chemicals (SMILES)

A standard sequential representation exists for chemicals, called Simplified Molecular-Input Line-Entry System (SMILES) and any other chemical format is convertible to it via open-source conversion means [8]. The chemical language SMILES was designed "*to represent molecular structure by a linear string of symbols, similar to a natural language*" [11] to facilitate computer storage and processing of chemicals. A sequence in SMILES represents a molecular structure as a graph in the following way.

Atoms: Atoms are represented by their atomic symbols: C, Cl, N, O, etc. This is the only required use of letters in SMILES. Hydrogen atoms (H) are normally omitted, since valences make it clear where they are missing. For example, an atomic chain CCSCCCCC is depicted in Figure 1.A.

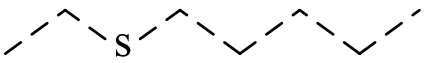


Fig. 1 A Atomic Chain CCSCCCCC

Bonds: Single bonds are usually omitted in SMILES. Double and triple bonds are represented by the symbols = and #, respectively, for example, in Figure 1.B.

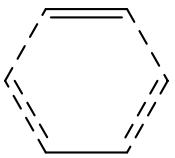


Fig. 1 B Double Bond. C1=CC=CC=C1.

Branches: Branches are specified by enclosures in parentheses, as in Figure 1.C.

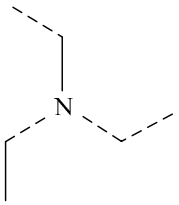


Fig. 1 C Branches CCN(CC)CC.

Cyclic Structures: Cyclic structures are represented by breaking one single (or aromatic) bond in each ring. The bonds are numbered in any order, designating ring-opening (or ring-closure) bonds by a digit immediately following the atomic symbol at each ring closure. This leaves a connected noncyclic graph, which is written as a noncyclic structure, as in Figure 1.D.

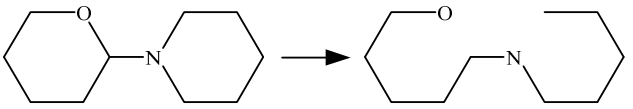


Fig. 1 D. Rings are broken, and a number is put to leave a mark where the bond was broken: O1CCCCC1N1CCCCC1.

With the rules above almost all organic structures can be described as strings. For more details, the reader is referred to the introduction into the SMILES [11].

3 Statistical Adapter

A flow chart of the Statistical Adapter is depicted in Figure 2. For the details on parsing and grammar inference, the reader is referred to [9,10].

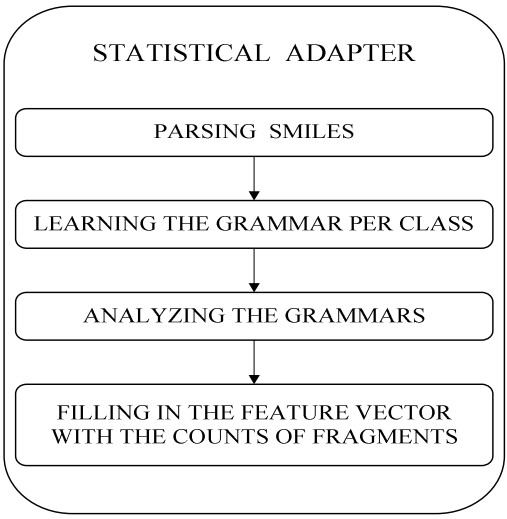


Fig. 2 The flow chart of the Statistical Adaptor.

- **INPUT:** The requirement on the input is that it is parsable. SMILES sequences are parsed into a syntax tree with our parser [10].
- **BLOCK 1: Learning grammars.** The general motivation is that the object's structure accounts for its properties, and that objects with similar structures have similar properties. Given two sets of examples from the opposite classes (for example, toxic and nontoxic chemicals), two grammars are learnt that summarize the structures. Examples are taken from the training set one by one. Whenever an example cannot be parsed with the current grammar, the grammar is extended with new rules to accommodate the example. Additionally, for each production, it is registered how many times it was used in the grammar learning process.
- **BLOCK 2: Analyzing the grammars.** *What are the interesting fragments and how should they be expressed with terminal and/or non-terminal symbols?* This issue is resolved by the grammar. Consider the examples of grammar rules from parsing:

$$sig_2 \rightarrow sig_6 sig_6,$$
$$sig_6 \rightarrow C1Csig_3CCC1,$$
$$sig_6 \rightarrow C1CCCCC1.$$

The non-terminal on the left-hand side of the rules entirely depends on the right hand side and is redundant, namely, it is of the form sig_{arity}, where sig means 'non-terminal' and the arity is the number of units that appear on the right-hand side. Thus, we can work with the right-hand side only. The grammar defines how the fragments are segmented and the level of abstraction (the use of non-terminals). In this example, the fragments are $sig_6 sig_6$, $C1Csig_3CCC1$, C1CCCCC1.

- **BLOCK 3: Filling in the feature vector.** The counts of these fragments in a molecule become candidates to be included as vector features, and the decision for inclusion or not will be taken via statistical feature selection.
- **OUTPUT:** The Adapter's output is a file in the comma-separated format (*.csv*), in which each line is a vector representation for a molecule from the input database.

4 Sequence Retriever

While the Adapter performs the conversion: sequence → vector with the counts of interesting molecular fragments, the Retriever performs the mapping: molecular fragment → sequences from the database that contain this fragment.

The Retriever takes two parameters: one or several fragments to search for and the database, where to search. It parses the molecules from the database and stores them as a collection of parsed units stored together with the sample Id from which the fragment was parsed. The molecules containing the fragment are found via a search in the first field in the data structure, and the Id points back to the chemical. The Sequence Retriever works in two modes:

1) it retrieves a list of molecules that contain some molecular fragments, and

2) it generates fingerprints[3] for molecules with the counts (or Boolean indicators for presence) of the specified fragments. The fragments can be:

1. **known**, for example, functional groups have been implemented (that is, specific groups of atoms or bonds within molecules that are responsible for characteristic reactions),
2. **discovered** by the Adapter, or
3. **user-specified,** and in this case they should be parsable units and be specified in the style of the grammar rules of Section 3.

Example: The NCTRER DSSTOX database[4] contains the SMILES of toxic and non-toxic chemicals with their experimental toxicity values. In [10], the fragment [C@, sig3, sig6] was automatically segmented as relevant to the activity. We specified it for the Retriever and it returned the list of compounds from the database (Table 1), which contain it.

[3] Fingerprints are vectors of binary digits (bits) that represent the presence or absence of particular fragments in a molecule or the counts of how many times the substructure is contained. Comparing fingerprints is a standard way to determine the similarity between molecules.

[4] http://www.epa.gov/nheerl/dsstox/sdf_nctrer.html

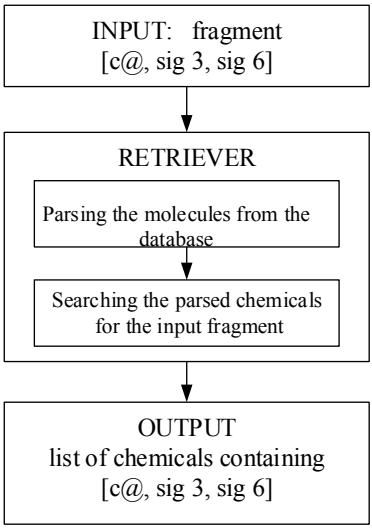


Fig. 3 The Flow Chart of the Sequence Retriever

Table 1 The retrieved molecules from NCTRER DSSTOX which contain the *C@ sig3 sig6*.

SMILES:	Activity-Value (toxicity):
OC1=CC=C(CCCCCCCC)C=C1	44
C1(=C(O)C=CC=C1)C(CC)C	30
C1(=CC=C(C=C1)O)C(CC)C	32

5 Discussion and Conclusions

For this submission, we have extended the Adapter for sequential input with the Sequence Retriever. The Sequence Retriever scans databases for different sorts of molecular fragments: known, discovered, or user-specified. The mechanism to define the fragments can express concrete fragments and the ones with variables. As a next step, we plan to add a drawing functionality to depict the fragments and implement the search with the help of a drawing.

The approach behind the Adapter and Retriever relies on diverse computational methods (parsing, grammar acquisition, statistical analysis of grammars, vector-based statistical predictive and feature selection methods), rather than on systematic scientific domain knowledge. Due to that, our method should be applicable in other applications dealing with sequential data including the ones outside bioinformatics and be used for data exploration and knowledge discovery purposes.

Acknowledgements The corresponding author is pleased to acknowledge being part of the *"Scalable resource-efficient systems for big data analytics"*, an industry-academia project at BTH funded by the Knowledge Foundation, Sweden.

References

1. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA data mining software: an update. ACM SIGKDD Explorations Newsletter **11**(1), 10–18 (2009)
2. Durbin, R., Eddy, S.R., Krogh, A., Mitchison, G.: Biological sequence analysis: probabilistic models of proteins and nucleic acids. Cambridge university press (1998). Chapter 3
3. Ernst, J., Kellis, M.: Discovery and characterization of chromatin states for systematic annotation of the human genome. Nature Biotechnology **28**(8), 817–825 (2010)
4. Arcas, A., Cases, I., Rojas, A.M.: Serine/threonine kinases and E2-ubiquitin conjugating enzymes in Planctomycetes: unexpected findings. Antonie van Leeuwenhoek **104**(4), 509–520 (2013)
5. Reverter, F., Vegas, E., Oller, J.M.: Kernel-PCA data integration with enhanced interpretability. BMC Systems Biology **8**(Suppl 2), S6 (2014)
6. Tetko, I.V., Gasteiger, J., Todeschini, R., Mauri, A., Livingstone, D., Ertl, P., Palyulin, V.A., Radchenko, E.V., Zefirov, N.S., Makarenko, A.S., Tanchuk, V.Y.: Virtual computational chemistry laboratory design and description. Journal of computer-aided molecular design **19**(6), 453–463 (2005)
7. Carbonell, P., Carlsson, L., Faulon, J.L.: Stereo signature molecular descriptor. Journal of chemical information and modeling **53**(4), 887–897 (2013)
8. OLBoyle, N.M., Banck, M., James, C.A., Morley, C., Vandermeersch, T., Hutchison, G.R.: Open Babel: An open chemical toolbox. J. Cheminf. **3**, 33 (2011)
9. Sidorova, J., Anisimova, M.: NLP-inspired structural pattern recognition in chemical application. Pattern Recognition Letters **45**, 11–16 (2014)
10. Sidorova, J., Garcia, J.: Bridging from syntactic to statistical methods: Classification with automatically segmented features from sequences. Pattern Recognition (2015)
11. Weininger, D.: SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. Journal of Chemical Information and Computer Sciences **28**(1), 31–36 (1988)
12. Tanaka, E.: Theoretical aspects of syntactic pattern recognition. Pattern Recognition **28**(7), 1053–1061 (1995)
13. Venguerov, M., Cunningham, P.: Generalised syntactic pattern recognition as a unifying approach in image analysis. In: Advances in Pattern Recognition, pp. 913–920. Springer, Heidelberg (1998)

Part II
Gene Expression

Sensitivity, Specificity and Prioritization of Gene Set Analysis When Applying Different Ranking Metrics

Joanna Zyla, Michal Marczyk and Joanna Polanska

Abstract Microarrays were a trigger to develop new methods which can allow to estimate disturbances in signal cascades, characterized by sets of genes, in various biological conditions. Existing approaches of gene set analysis take information if genes are differentially expressed or are based on some gene ranking. The most commonly used method is Gene Set Enrichment Analysis (GSEA), where an assumption of uniform distribution of genes in some gene set is tested by weighted Kolmogorov-Smirnov test. Many studies present different gene set analysis methods and their comparison, however none of them focus on basic but crucial parameters, like the rank metric. In this paper we compare nine ranking metrics in terms of sensitivity, specificity and prioritization of identification of functional gene sets using a collection of 34 annotated microarray datasets. We show that absolute value of default GSEA measure is the best ranking metric, while the Baumgartner-Weiss-Schindler test statistic is the best statistical-based metrics, which can be used in Gene Set Enrichment Analysis.

Keywords Gene set analysis · Ranking metrics · Functional enrichment efficiency

1 Introduction

From the moment of introducing microarrays into molecular biology, they became one of the most popular methods to identify differentially expressed genes (DEGs) under many diseases and conditions. Microarray experiments can show hundreds or thousands DEGs, which cause biological interpretation of results to be hard.

J. Zyla(✉) · M. Marczyk · J. Polanska
Data Mining Group, Institute of Automatic Control, Faculty of Automatic Control, Electronics and Computer Science, Silesian University of Technology, Akademicka 16, 44-100 Gliwice, Poland
e-mail: {joanna.zyla,michal.marczyk,joanna.polanska}@polsl.pl

M.S. Mohamad et al. (eds.), *10th International Conference on PACBB*, Advances in Intelligent Systems and Computing 477,
DOI: 10.1007/978-3-319-40126-3_7

On the other hand, they can show no DEGs, which may be caused by low power of used methods. Efficiency of searching for DEGs may be improved by gene filtering that is based on eliminating genes that most probably do not differ in expression between different experimental conditions e.g. by summarized expression statistics, like average gene expression [1]. Unfavorable results of finding DEGs put enormous impact to the development of methods which can point out the overrepresented or enriched group of genes (called gene set) e.g. KEGG pathway or Gene Ontology. The growing popularity of gene set analysis method reached the point, where almost every bioinformatics study looks for significant pathways as explanation of biological processes or validation of computationally derived results.

Gene set analysis methods can be divided into three main types. First group of existing methods is called Over-Representation Analysis (ORA) [2], also known as a first generation approach. They are based on analysis of contingency tables constructed from number of DEGs and non-DEGs. The hypergeometric test with chi-square distribution is used to establish significance of each gene set. A serious drawback of ORA methods is that it cannot be applied if no DEGs are found on given significance level. Also the hypergeometric test assumption about gene expression independency is in most of the cases not fulfilled. This constitutes serious drawback of ORA which treats each gene equally without looking for information about the extent of regulation (fold-changes, significance of a change, etc.). To solve these problems Subramanian et al. proposed a method called Gene Set Enrichment Analysis (GSEA) [3], which still is the most commonly used. The GSEA method is based on gene ranking list used to derive enrichment score for all genes that belong to a given gene set, regardless of whether or not they are differentially expressed. Statistical significance of each gene set is established by applying a permutation test. Similar methods were also proposed, like: Pathway Analysis with Down-weighting of Overlapping Genes (PADOG) [4] or Correlation Adjusted Mean Rank gene set test (CAMERA) [5]. This group of methods of gene set investigation is called Functional Class Sorting (FCS), known also as a second generation methods. Last group is known as Pathway Topology (PT)-based approaches (third generation). In structure they are similar to FCS methods, but they use pathway topology to compute gene-level statistics. In this group methods like ScorePAGE [6] or NetGSEA [7] were proposed.

Despite the numerous number of proposed gene set analysis methods and their comparisons and explanations, like work of Hung et al. [8] or Maciejewski [9], still there exist some challenges reviewed by Khatri et al. [10], like incomplete annotation or lack of benchmark data sets. Solution for last problems were proposed by Tarca et al. [11]. Nevertheless, in case of rank metrics, study on large dataset was not performed before. In present work we show how different ranking of genes in GSEA influences the outcome using overall statistical performance measures like sensitivity, specificity and prioritization. To sum up, the conducted study is based on a large collection of microarray datasets and it was never investigated before in case of GSEA ranking metrics methods.

2 Material

In the presented study, microarray datasets that are publicly available in R Bioconductor were used. First package used is KEGGdzPathwaysGEO [4]. It consists of 24 microarray datasets, which were pre-processed in the following way: a) Affymetrix arrays were normalized by RMA algorithm giving log2 expressions, b) Illumina arrays were normalized using the quantile normalization algorithm giving log2 expressions, c) duplicates were removed by keeping the probeset with smallest p-value. Second collection of data is KEGGandMetacoreDzPathwaysGEO package, which consists of 18 microarray datasets. All data within that package were normalized by RMA algorithm, giving log2 expressions and the duplicates were removed by keeping the probeset with the highest average expression across all samples. Due to the lack of access to Metacore resources only datasets with KEGG identifier were used. The final set of 34 microarray collections was firstly analyzed by checking normality of gene expression distribution (Lilliefors test) and homogeneity of variance between phenotypes (F-test). The KEGG pathways gene lists were obtained via the KEGGREST Bioconductor package giving 273 different pathways [12].

3 Methods

The Gene Set Enrichment Analysis (GSEA) method proposed by Subramanian et al. is used to test different ranking methods [3]. The general idea of the method is to check if the distribution of genes (according to established rank) in the gene set differs from a uniform distribution using a weighted Kolmogorov-Smirnov test. The main measure in GSEA method is the Enrichment Score (ES), which is described as maximum deviation from zero between hits of genes into gene set *S*, marked as P_{hit} (Eq.1) and hits of genes outside gene set, marked as P_{miss} (Eq. 2).

$$P_{hit}(S,i) = \sum_{\substack{g_j \in S \\ j \le i}} \frac{|r_j|^p}{N_R} \quad \text{where} \quad N_R = \sum_{g_j \in S} |r_j|^p \tag{1}$$

$$P_{hit}(S,i) = \sum_{\substack{g_j \notin S \\ j \le i}} \frac{1}{N - N_H} \quad \text{where} \quad N_H \quad \text{no. of genes in Gene Set “S”} \tag{2}$$

To assess the significance of an observed ES the permutation test is performed. In GSEA method two types of permutations can be made: by phenotypes and by gene labels. The latter is under consideration in this work. To adjust for variation in gene set size the normalized enrichment score (NES) is calculated, which is the main measure of gene sets functional enrichment used in this study. The weight parameter p from Eq.2 was set to 1 which causes rank metric r the main influence parameter to the final result. Such setting is default and recommended by GSEA authors.

In the presented study, the nine ranking metrics are tested. The first group of ranking metrics consists of those available in GSEA java application (http://software. broadinstitute.org/gsea/downloads.jsp): signal-to-noise ratio (S2N) – default measure, absolute value from signal-to-noise ratio (abs(S2N)), difference of expression means between classes (Difference), ratio of expression means of two classes (Ratio) and its log2 scale (log2(Ratio)), T-test statistic (T-test) and its absolute value (abs(T-test)). Existing ranking metrics are expanded by two non-parametric metrics proposed in Baya et al. [13]: Wilcoxon statistic (Wilcoxon), and Baumgartner-Weiss-Schindler test statistic (BWS). The Wilcoxon method is based on Wilcoxon Rank Sum test, where there is no assumption about the probability distribution of expression data. It is more robust in case of existing outliers than parametric methods, like T-test. The Wilcoxon statistic is computed as the sum of the ranks for the first class. The BWS test is based on the squared value of the difference between the two empirical distribution functions weighted by the respective variance and approximated by average of B statistics of each class. Neuhäuser showed that BWS gives more accurate Type I error control and more power as compared to the Wilcoxon test [14].

For each ranking metric three measures of enrichment performance are established: i) prioritization – selecting rank close to the top of gene set list that are indeed relevant to a given problem, ii) sensitivity – production of small p-values of NES statistic for relevant gene sets, and iii) specificity – generating false positives no more than expected. Those three measures and microarray data were introduced in Tarca et al. [9] to compare different methods of gene-set analysis, including GSEA. The GSEA method and all measures were implemented in MATLAB R2015b software.

4 Results and Discussion

First step of the analysis was to check normality of gene expression distributions and homogeneity of expression variances between classes. In the analyzed datasets expressions of most of the genes were distributed non-normally and variances were non-homogeneous (no. of genes where H_0 was rejected was larger 5% of all). For each dataset sensitivity was represented by p-value of NES statistic of target pathway, calculated using 10,000 permutations of genes labels. The tied rank was assign to the position of target pathway in gene set and normalized to assess prioritization. The results for sensitivity and prioritization for all 34 datasets are presented on boxplots at Fig. 1 and Fig. 2, respectively. Each box represents distribution of p-values or normalized position in the gene set list of target pathway across all tested datasets. In both figures horizontal lines across all boxplots represent theoretical sensitivity and prioritization for random gene expression experiment.

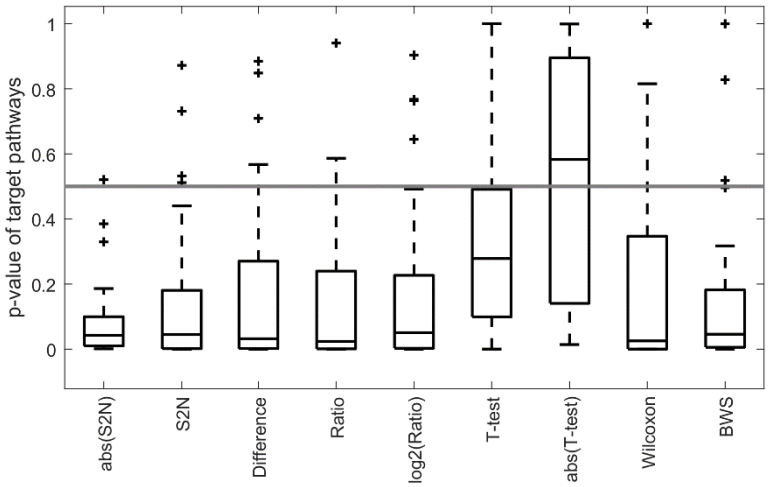


Fig. 1 Boxplots of sensitivity measure for all analyzed ranking metrics to each dataset and its dedicated pathway. Lower values show higher sensitivity.

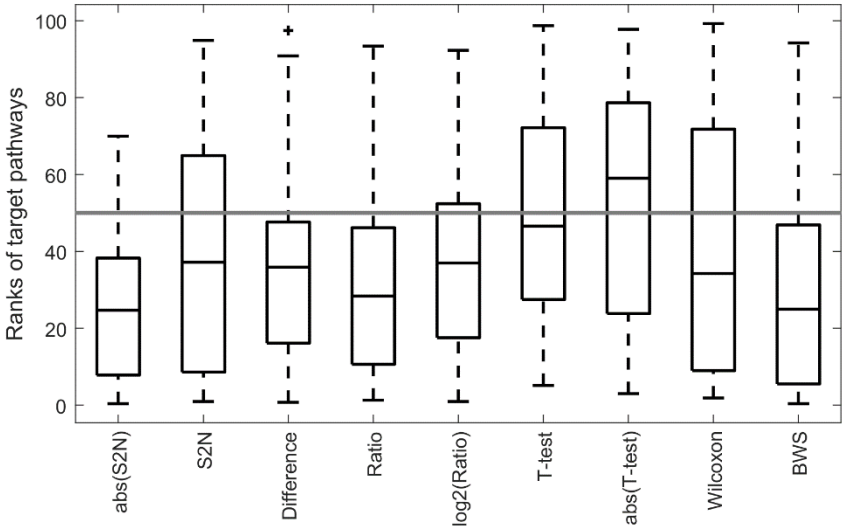


Fig. 2 Boxplots of prioritization measure for all analyzed ranking metrics to each dataset and its dedicated pathway. Lower values show higher prioritization.

Overall sensitivity and prioritization are summarized by median statistic. As can be seen from the median of NES p-values Ratio metric gives more sensitive outcomes than others. By evaluating interquartile range of sensitivity, we can say that abs(S2N) method gives the most reproducible results. Advantage of abs(S2N) method over S2N relies on capability of enriching gene sets, where expression

change between classes in dependent genes is not in the same direction (some genes are significantly up- and some down-regulated). Among the statistical ranking metrics (t-test, abs(t-test), Wilcoxon and BWS) the best results are observed for BWS method. It shows the power of the non-parametric metrics to find proper gene set increase in case of non-normal distribution of data comparing to parametric t-test. Most non-statistical methods have median value of sensitivity lower than 0.5, but among statistical measures only BWS and Wilcoxon rank metric gives sensitivity higher than by chance. This situation is not so clear in case of prioritization, where large deviations for each ranking metric are observed. The best outcomes for prioritization measure are observed for abs(S2N) method. Again, BWS metric gives median prioritization much lower than a random level of 50% among statistical-based methods. Last analysis measure was specificity for which 50 permutations of phenotypes were performed as suggested by Tarca et al. [9], and the number of gene sets with NES p-value lower than 0.05 was collected. P-value of NES statistic was calculated using 1,000 permutations of genes within gene sets. Results of this step are presented in Fig. 3 where horizontal grey line represents the level of false positives for statistical significance set to 0.05.

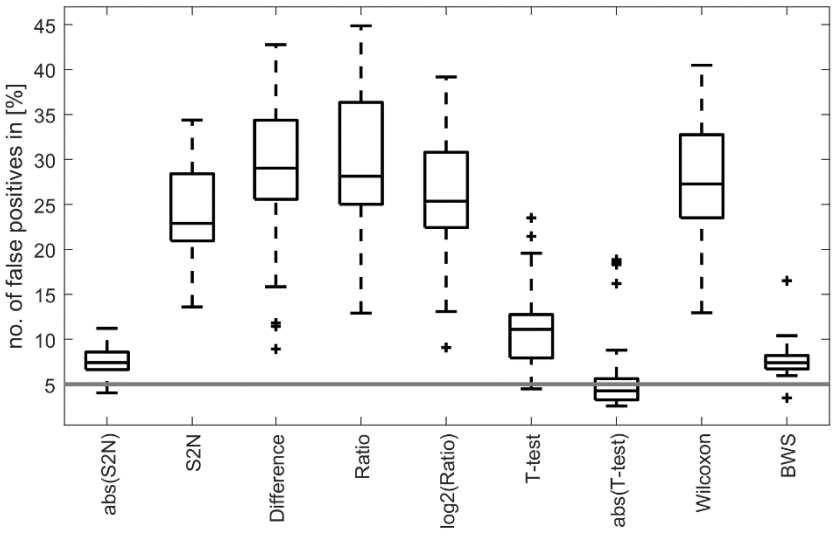


Fig. 3 Boxplots of specificity measure for all analysed ranking metrics and all datasets permuted by phenotype 50 times.

As can be observed the specificity, when 5% significance level was set, is the best for abs(T-test) method. Abs(S2N) method gives the best outcomes for non-statistical methods. Additionally, BWS shows similar results to abs(S2N). Also the range of results is narrow, which shows stability of this rank metric. Finally, except abs(T-test), BWS and abs(S2N), most of method shows very weak specificity caused by gene-set permutation type. This situation should be more stable when performing phenotype permutation in GSEA.

Table 1 Basic characteristics of metrics in each comparison measure. Bold values represent statistically significant outcomes. Grey color represent statistical-based ranking metrics.

Metric	Med. Sens.	Med. 95% CI	Med. Priorit.	Med. 95% CI	Mean Spec.	Mean 95% CI
abs(S2N)	0.042	**[0.00; 0.09]**	24.676	**[16.14; 33.21]**	7.632	[7.13; 8.14]
S2N	0.044	**[0.00; 0.14]**	37.179	[24.18; 50.18]	24.273	[22.45; 26.10]
Difference	0.032	**[0.00; 0.14]**	35.897	**[23.59; 48.20]**	29.121	[26.23; 32.01]
Ratio	0.023	**[0.00; 0.12]**	28.388	**[17.15; 39.62]**	29.617	[26.91; 32.32]
log2(Ratio)	0.050	**[0.00; 0.16]**	36.996	**[24.89; 49.11]**	25.503	[23.02; 27.98]
T-test	0.278	**[0.15; 0.40]**	46.564	[34.75; 58.38]	11.532	[10.08; 12.98]
abs(T-test)	0.583	[0.42; 0.75]	59.033	[45.62; 72.45]	5.947	**[4.32; 7.58]**
Wilcoxon	0.025	**[0.00; 0.15]**	34.249	**[20.45; 48.04]**	28.079	[25.78; 30.38]
BWS	0.046	**[0.00; 0.15]**	24.948	**[13.30; 36.59]**	7.714	[7.06; 8.41]

Table 2 Ranking of gene set analysis methods. Z-scores of each value from Table 1 was calculated and sum as final estimation for each ranking metrics. Grey color represent statistical-based ranking metrics. Bold font represent the best result in each group of measures.

Metric	Z-score			
	Sensitivity	Prioritization	Specificity	**Sum**
abs(S2N)	-0.44	**-1.08**	-1.09	-2.60
BWS	-0.42	-1.05	-1.08	-2.55
Ratio	**-0.54**	-0.74	1.05	-0.23
Wilcoxon	-0.53	-0.20	0.90	0.17
S2N	-0.42	0.07	0.53	0.17
log2(Ratio)	-0.39	0.05	0.65	0.30
Difference	-0.49	-0.05	1.00	0.46
T-test	0.81	0.93	-0.71	1.03
abs(T-test)	2.42	2.07	**-1.25**	3.24

To summarize the comparison of different GSEA ranking metrics, the median values of sensitivity and prioritization were gathered together with mean specificity calculated at 5% significance level (boxplot shows not-skewed distribution so mean value will be suitable as representative). The 95% Confidence Interval (CI) was calculated for each evaluation metric (sensitivity, prioritization and specificity). Results are presented in Table 1, where the bold values represent metrics where CI do not include random estimation for sensitivity and prioritization and include 5% for specificity. Further, each group of results was standardized by z-transformation and summed up to find the best rank method for gene set enrichment analysis, where the lower value will show the best rank metrics. Results of this analysis are presented in Table 2. Final results presented in Table 2 are sorted from lowest sum of z-scores value to the highest (lower value shows better overall performance). The best ranking metric approach from nine tested methods

is abs(S2N), which gives the best prioritization, while specificity and sensitivity are at satisfactory high level. Very similar outcomes are observed for statistical-based metrics BWS. What is worth to underline the Wilcoxon method gives better sensitivity comparing to abs(S2N) and BWS. However, the weakest point of Wilcoxon metrics is low specificity, which gives too many falsely enriched gene sets.

5 Conclusions

The ability to give sensitive, specified and high prioritized results in term of ranking metric in Gene Set Enrichment Analysis with gene set permutation was shown. Based on total performance the abs(S2N) method appears to be the best ranking metric from nine tested. However, in case of abs(S2N), without additional permutation test it is hard to show which genes are differentially expressed. From the group of statistical-based methods, the BWS metric gives the best overall performance (very close to abs(S2N)), which in case of non-normally distributed expression data is the expected outcome. Additionally, it shows recently described better power of BWS method over Wilcoxon test [14], also in case of GSEA. In the presented study only the basic gene ranking metric methods are investigated and extension to other feature selection methods is needed. Finally, the presented research is unique in case of number of analyzed microarray datasets and statistical measures used to assess the best ranking metrics. It was shown that not default GSEA rank metrics but its absolute value is the best rank metric with overall estimation close to BWS statistic, not often used before.

Acknowledgements This work was financed by NCN grant HARMONIA 4 no. 2013/08/M/ST6/00924 (MM, JP) and grant no. BKM/514/RAU1/2015/t.20 (JZ). All the calculations were carried out using infrastructure funded by GeCONiI project (POIG.02.03.01-24-099/13).

References

1. Marczyk, M., Jaksik, R., Polanski, A., Polanska, J.: Adaptive filtering of microarray gene expression data based on Gaussian mixture decomposition. BMC Bioinformatics **14**(1), 101 (2013)
2. Tavazoie, S., Hughes, J.D., Campbell, M.J., Cho, R.J., Church, G.M.: Systematic determination of genetic network architecture. Nature Genetics **22**(3), 281–285 (1999)
3. Subramanian, A., et al.: Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. PNAS **102**(43), 15545–15550 (2005)
4. Tarca, A.L., Draghici, S., Bhatti, G., Romero, R.: Down-weighting overlapping genes improves gene set analysis. BMC Bioinformatics **13**, 136 (2012)
5. Wu, D., Smyth, G.K.: Camera: a competitive gene set test accounting for inter-gene correlation. Nucleic Acids Research **40**(17), e133 (2012)

6. Rahnenführer, J., Domingues, F. S., Maydt, J., Lengauer, T.: Calculating the statistical significance of changes in pathway activity from gene expression data. Statistical Applications in Genetics and Molecular Biology **3**(1) (2004)
7. Shojaie, A., Michailidis, G.: Network enrichment analysis in complex experiments. Statistical Applications in Genetics and Molecular Biology **9**(1) (2010)
8. Hung, J.-H., et al.: Gene set enrichment analysis: performance evaluation and usage guidelines. Briefings in Bioinformatics **13**(3), 281–291 (2012)
9. Maciejewski, H.: Gene set analysis methods: statistical models and methodological differences. Briefings in Bioinformatics **15**(4), 504–518 (2014)
10. Khatri, P., Sirota, M., Butte, A.J.: Ten years of pathway analysis: current approaches and outstanding challenges. PLoS Computational Biology **8**(2), e1002375 (2012)
11. Tarca, A.L., Bhatti, G., Romero, R.: A comparison of gene set analysis methods in terms of sensitivity, prioritization and specificity. PLoS One **8**(11), e79217 (2013)
12. Kanehisa, M., et al.: KEGG as a reference resource for gene and protein annotation. Nucleic Acids Research **44**(D1), D457–D462 (2016)
13. Bayá, A.E., Larese, M.G., Granitto, P.M., Gómez, J.C., Tapia, E.: Gene set enrichment analysis using non-parametric scores. In: Advances in Bioinformatics and Computational Biology, pp. 12–21. Springer, Heidelberg (2007)
14. Neuhäuser, M.: An exact two-sample test based on the Baumgartner-Weiß-Schindler statistic and a modification of Lepage's test. Communications in Statistics-Theory and Methods **29**(1), 67–78 (2000)

Deep Data Analysis of a Large Microarray Collection for Leukemia Biomarker Identification

Wojciech Labaj, Anna Papiez, Joanna Polanska and Andrzej Polanski

Abstract Nowadays, statistical design of experiments allows for planning of complex studies while maintaining control over technical bias. In this study, the equal importance of performing tailored preprocessing, such as batch effect adjustment and adaptive signal filtration, is demonstrated in order to enhance quality of the results. This approach is assessed on a large set of data on acute and chronic leukemia cases. It is shown, both through statistical analysis and literature research, that drawing attention toward data preprocessing is worthwhile, as it produces meaningful original biological conclusions. Specifically in this case, it entailed the revealing of four candidate leukemia biomarkers for further investigation of their significance.

Keywords Batch effect · Leukemia · Biomarker identification · Gene expression · High-throughput

1 Background

To this date, a great understanding of the importance for the methods of experimental design has been developed within the scientific community. The principles of control, replication and randomization are commonly known and implemented throughout laboratories and research institutions regardless of the study field. This considerate approach allows for the planning of gradually more complex experiments with higher

W. Labaj(✉) · A. Polanski
Silesian University of Technology, Institute of Informatics, Akademicka 2A,
44-100 Gliwice, Poland
e-mail: {wojciech.labaj,andrzej.polanski}@polsl.pl

A. Papiez · J. Polanska
Silesian University of Technology, Institute of Automatic Control, Akademicka 2A,
44-100 Gliwice, Poland
e-mail: {anna.papiez,joanna.polanska}@polsl.pl

M.S. Mohamad et al. (eds.), *10th International Conference on PACBB*,
Advances in Intelligent Systems and Computing 477,
DOI: 10.1007/978-3-319-40126-3_8

power of statistical testing and thus, a better chance of obtaining meaningful novel results. As an example, the Microarray Innovations in Leukemia research experiment [5] has been established and carried out with a great deal of state-of-the-art protocols and strict control procedures during the experimental stage. On the basis of this study, the presented work aims to show how equally careful and tailored data preprocessing is vital to the final investigation outcome and conclusions and how it should be as commonly inconceivable to neglect this essential step in biomedical data mining.

2 Materials and Methods

2.1 Data Sets

The Microarray Innovations in Leukemia (MILE) study [5] was designed to assess the clinical accuracy of gene expression profiles, originating from microarray experiments, compared to standard leukemia laboratory methods ("Gold Standard") for 16 acute and chronic leukemia subclasses, myelodysplastic syndromes (MDSs) and control group that included nonmalignant disorders and normal bone marrow. The leukemia subclasses may be divided into four main groups: acute and chronic myeloid leukemia (AML, CML) and acute and chronic lymphoblastic leukemia (ALL, AML). The investigation was performed in 11 laboratories across three continents and included a total of 3, 334 patients. The study was very carefully designed to eliminate main problems which occur when many experiments are carried out in various laboratories by different laboratory technicians - so called *batch effect*. The study consisted of four phases: two main phases (Stage I and Stage II), each of them preceded by a pre-phase [7]. The goals of the pre-phases were to assure intralaboratory reproducibility and interlaboratory comparability. Each laboratory operator was trained on an identical sample preparation protocol. Additionally, each laboratory was provided with the same laboratory equipment and also kits and reagents for sample preparation and microarray analysis were taken from the same source.

In this analysis microarray data from Stage I of the MILE study were investigated, where 2, 096 bone marrow samples of acute and chronic leukemia patients were hybridized to Affymetrix HG-U133 Plus 2.0 GeneChips. Summary of the MILE datasets Stage I is presented in Table 1.

2.2 Preprocessing

The intensity data from microarray experiments has been subject to fRMA normalization [11] with background correction, quantile normalization and median polish summarization. This method has been chosen to merge the advantages of classic

Table 1 Summary of the MILE datasets (STAGE I). Types of leukemia defined by gold standard methods by the participating institutions.

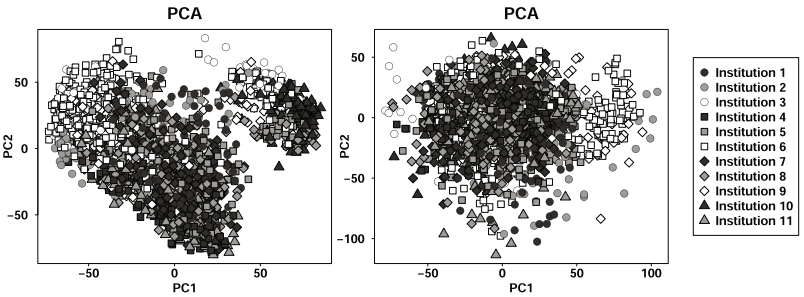
Class	Diagnosis	Total N^o of samples
C1	Mature B-ALL with t(8;14)	13
C2	Pro-B-ALL with t(11q23)/MLL	70
C3	c-ALL/pre-B-ALL with t(9;22)	122
C4	T-ALL	174
C5	ALL with t(12;21)	58
C6	ALL with t(1;19)	36
C7	ALL with hyperdiploid karyotype	40
C8	c-ALL/pre-B-ALL without t(9;22)	237
C9	AML with t(8;21)	40
C10	AML with t(15;17)	37
C11	AML with inv(16)/t(16;16)	28
C12	AML with t(11q23)/MLL	38
C13	AML with normal karyotype + other abnormalities	351
C14	AML complex aberrant karyotype	48
C15	CLL	448
C16	CML	76
C17	MDS	206
C18	Non-leukemia and healthy bone marrow	74
	Total	2,096

Table 2 Results of Kruskal-Wallis test for gene differentiation among research centers participating in sample preparation and leukemia subgroups ($\alpha = 0.05$).

	N^o of genes	
	Research centers	Leukemia subtype
Total	9, 941	
No batch effect correction	9, 939	9, 925
After batch effect correction	130	9, 851

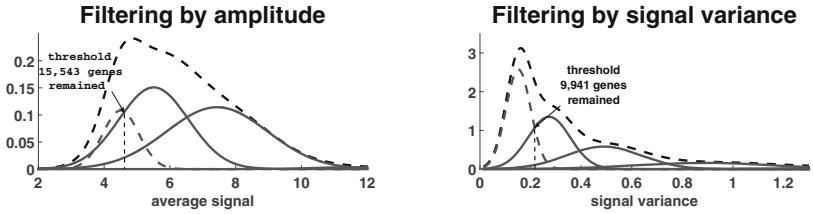
RMA normalization with the ability to include additional samples if need in the future. Probe reannotation was accomplished with custom CDF files available through the BrainArray repository [1].

The next step was to ensure data coherence, i.e. verify if the unification procedures applied in the study successfully dealt with the issue of bias introduced



(a) No batch effect correction. (b) After batch effect correction.

Fig. 1 Principal component graphs demonstrating the existence of batch effect in the data with regard to sample preparation research center.



(a) Gaussian mixture decomposition by signal amplitude. (b) Gaussian mixture decomposition by variance.

Fig. 2 Decomposition into Gaussian components as a method of filtration of genes with signal intensity close to background values and low variance.

by batch effect. In this case Principal Component Analysis was performed and the outcome suggests that nonetheless a batch effect due to sample preparation in different laboratories may be observed (Figure 1).

Therefore, the data were adjusted for batch effects with the use of ComBat algorithm [6], available through the SVA R package [9]. The results of Kruskall-Wallis test for differentially expressed genes among research center batches proved a significant removal of batch effect (Table 2).

The final preprocessing step consisted of gene filtration in order to remove features with signal close to background level. There are various techniques available for this purpose such as the commonly used method of removing 50% of the genes with lowest expression value or variance [4]. However, in the studied case of 18 subtypes of disease this approach seems excessively strict and implies the search of an adaptive threshold rather than fixed. For this reason, the adaptive filtering based on Gaussian mixture decomposition has been selected [10]. The filtration was conducted in two steps: in the first stage the signal was decomposed in terms of signal intensity amplitude, and the three components with the highest signal amplitude

remained. Secondly, the data were considered variance-wise and the component with lowest variance was rejected (Figure 2). A total of 9, 941 genes remained for further statistical analysis.

2.3 Statistical Analysis and Biomarker Selection

In order to search for features differentially expressed across subtypes of leukemia analysis of variance tests were carried out. After verifying that the conditions of normality and symmetry of the distribution do not hold, the non-parametric Kruskal-Wallis analysis of variance test was performed [8].

Furthermore, as means of conducting post-hoc pairwise comparison tests, the Games-Howell method was chosen [3]. Restrictive feature selection was then used in order to filter out the genes which differentiate solely one group from all of the other subtypes of leukemia. The preprocessing steps and feature selection method form an innovative pipeline for deep expression data analysis (Figure 3).

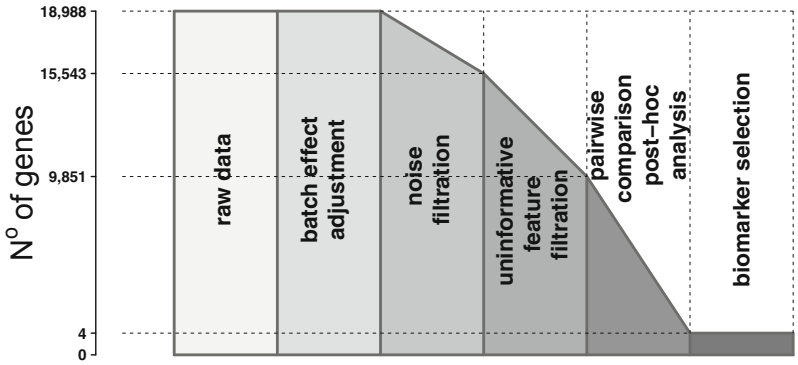


Fig. 3 Expression data analysis pipeline with gradual gene reduction.

3 Results

3.1 Statistical Analysis

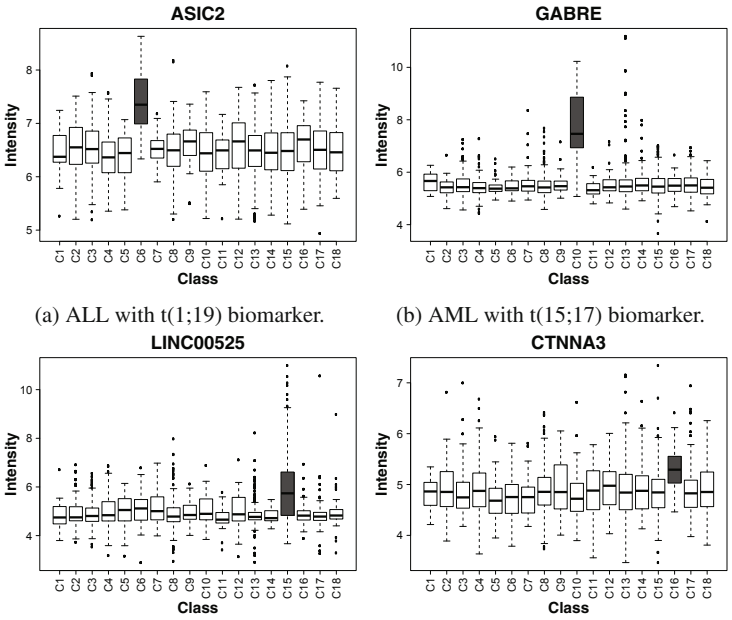
The Kruskal-Wallis test results demonstrate that an overwhelming majority of the genes remaining for analysis present differentiation between the studied subgroups of leukemia (Table 2). This is foreseeable as with such a large number of groups after adequate gene filtration it is highly probable that at least one type will vary from the others significantly. Thus, the pairwise comparisons were carried out between the subgroups and the final results (Table 3) pointed out to merely four genes (one for a

single subgroup in each group of leukemia: AML, ALL, CML, CLL) differentiating a subgroup from all the others. The genes mentioned are (Figure 4): (1) ASIC2 - acid sensing ion channel 2, (2) GABRE - gamma-aminobutyric acid A receptor, epsilon, (3) LINC00525 - long intergenic non-protein coding RNA 525, (4) CTNNA3 - catenin alpha 3.

Table 3 Results of Games-Howell post-hoc pairwise comparisons. The subtype biomarkers are genes which differentiate only a particular subtype of disease from all the other subclasses.

Leukemia subtype	C1	C2	C3	C4	C5	C6	C7	C8	C9
N^o of differentiating genes	11	46	18	111	136	90	105	2	27
Leukemia subtype biomarkers	0	0	0	0	0	1	0	0	0

Leukemia subtype	C10	C11	C12	C13	C14	C15	C16	C17	C18
N^o of differentiating genes	141	36	19	4	3	18	90	1	2
Leukemia subtype biomarkers	1	0	0	0	0	1	1	0	0



(a) ALL with t(1;19) biomarker.

(b) AML with t(15;17) biomarker.

(c) CLL biomarker.

(d) CML biomarker.

Fig. 4 Boxplots illustrating the biomarker gene expression value distributions for subclasses of leukemia.

3.2 Biomarker Research

The genes which appeared to be candidate biomarkers for the different subtypes of disease have been investigated toward literature references related to leukemia. The CTNNA3 gene has been shown to be linked to the Shwachman-Diamond syndrome which is characterized by a high risk of leukemia [2]. In terms of relation to the bone marrow processes the GABRE gene which is a gammaaminobutyric acid receptor has proved to play a role during bone marrow stromal cell transplantation in the injured spinal cord in mice [12]. The remaining features did not prove to have literature connotations with leukemia and related subjects.

4 Discussion

The analyzed data originate from one of the main phases of MILE study and contain 2,096 samples, which were prepared by 11 research centers from around the world. This may be the cause of impairment of the quality of data by the impact of technical factors related to each research center. However, the whole experiment was very well designed, which means every laboratory was provided with the same equipment, kits, reagents coming from a common manufacturer or source and also the technicians were prepared in terms of using identical sample preparation protocol. Therefore, the data should not have been greatly affected by bias.

The analysis, which was adapted to the specific nature of the analyzed data, revealed that despite a well designed experiment, variability exists in the data associated with sample preparation by particular research institutes. Therefore, the data were adjusted for batch effects. It was presented both in the illustration of first and second PCA component and also by analysis of variance using Kruskal-Wallis test for research institutions, before and after batch effect correction. This study indicates that batch effect correction should be an indispensable element of microarray analysis protocol, because often it is impossible to exclude the impact of all external factors.

Targeted analysis of the data, which comprised fRMA, custom CDF file reannotation, batch effect correction, filtering using a method with adaptive selection of threshold and an appropriate set of statistic tests, allowed the extraction of important information. Four genes were discovered (ASIC2, GABRE, LINC00525, CTNNA3), which may be biomarkers for four subtypes of leukemia (ALL with t(1;19), AML with t(15;17), CML, CLL), one for each of the four major groups (AML, ALL, CML, CLL).

Some of the genes, which were discovered as a leukemia biomarker after our analysis, are described in the literature. Information, which has been found in the course of literature research, coincides to some extent with information about CTNNA3 and GABRE gene involvement in branches of diseases associated with leukemia. However, the discovered ASIC2 and LINC00525 biomarkers are not mentioned in the literature in this context.

5 Conclusions

The presented research demonstrated the significance of thorough data preprocessing including batch effect adjustment and adaptive filtration for inference in a well designed large study of gene expression data in leukemia patients. The above has been confirmed through statistical analysis as well as literature survey of the biological conclusions and resulted in finding four candidate biomarkers.

These preliminary results imply further investigation by means of performing classification analysis with the use of obtained differentiating features. The unique candidate biomarkers that have not been previously described in literature require experimental assessment in order to ultimately validate their suitability as auxiliary indicators of disease subtypes in leukemia.

Acknowledgments This work was financially supported by SUT grants BKM/515/RAU-2/2015 (WL), BKM/514/RAU-1/2015 (AP), NCN grant HARMONIA 4 2013/08/M/ST6/00924 (JP). The computations were carried out using GeCONiI infrastructure POIG.02.03.01-24-099/13.

References

1. Dai, M., Wang, P., Boyd, A.D., Kostov, G., Athey, B., Jones, E.G., Bunney, W.E., Myers, R.M., Speed, T.P., Akil, H., et al.: Evolving gene/transcript definitions significantly alter the interpretation of GeneChip data. Nucleic Acids Research **33**(20), e175 (2005)
2. Dhanraj, S., Manji, A., Pinto, D., Scherer, S.W., Favre, H., Loh, M.L., Chetty, R., Wei, A.C., Dror, Y.: Molecular characteristics of a pancreatic adenocarcinoma associated with Shwachman-Diamond syndrome. Pediatric Blood & Cancer **60**(5), 754–760 (2013)
3. Games, P.A., Howell, J.F.: Pairwise multiple comparison procedures with unequal N's and/or variances: a Monte Carlo study. Journal of Educational and Behavioral Statistics **1**(2), 113–125 (1976)
4. Hackstadt, A.J., Hess, A.M.: Filtering for increased power for microarray data analysis. BMC Bioinformatics **10**(1), 11 (2009)
5. Haferlach, T., Kohlmann, A., Wieczorek, L., Basso, G., Te Kronnie, G., Béné, M.-C., De Vos, J., Hernández, J.M., Hofmann, W.-K., Mills, K.I., et al.: Clinical utility of microarray-based gene expression profiling in the diagnosis and subclassification of leukemia: report from the International Microarray Innovations in Leukemia Study Group. Journal of Clinical Oncology **28**(15), 2529–2537 (2010)
6. Johnson, W.E., Li, C., Rabinovic, A.: Adjusting batch effects in microarray expression data using empirical Bayes methods. Biostatistics **8**(1), 118–127 (2007)
7. Kohlmann, A., Kipps, T.J., Rassenti, L.Z., Downing, J.R., Shurtleff, S.A., Mills, K.I., Gilkes, A.F., Hofmann, W.-K., Basso, G., DellOrto, M.C., et al.: An international standardization programme towards the application of gene expression profiling in routine leukaemia diagnostics: the Microarray Innovations in LEukemia study prephase. British Journal of Haematology **142**(5), 802–807 (2008)
8. Kruskal, W.H., Wallis, W.A.: Use of ranks in one-criterion variance analysis. Journal of the American Statistical Association **47**(260), 583–621 (1952)
9. Leek, J.T., Johnson, W.E., Parker, H.S., Jaffe, A.E., Storey, J.D.: The SVA package for removing batch effects and other unwanted variation in high-throughput experiments. Bioinformatics **28**(6), 882–883 (2012)

10. Marczyk, M., Jaksik, R., Polanski, A., Polanska, J.: Adaptive filtering of microarray gene expression data based on Gaussian mixture decomposition. BMC Bioinformatics **14**(1), 101 (2013)
11. McCall, M.N., Bolstad, B.M., Irizarry, R.A.: Frozen robust multiarray analysis (fRMA). Biostatistics **11**(2), 242–253 (2010)
12. Yano, S., Kuroda, S., Shichinohe, H., Seki, T., Ohnishi, T., Tamagami, H., Hida, K., Iwasaki, Y.: Bone marrow stromal cell transplantation preserves gammaaminobutyric acid receptor function in the injured spinal cord. Journal of Neurotrauma **23**(11), 1682–1692 (2006)

Cancer Detection Using Co-Training of SNP/Gene/MiRNA Expressions Classifiers

Reham Mohamed, Nagia M. Ghanem and Mohamed A. Ismail

Abstract Recent studies have explored using SNPs, miRNA and gene expression profiles for detection of different types of cancer. Many attempts have been proposed in literature to build machine learning classifiers for these genomic data types. However, these studies did not totally exploit the relations between the three data types. These studies also suffer from the scarcity of microArray labeled data. In this paper, we propose a new system for detecting cancer and its subtypes using co-training of SNPs, gene and miRNA classifiers. We leverage the relations between SNPs and genes, and the relations between miRNAs and genes, to predict one type of expression from the other. We also introduce a new method to predict the SNP microarray data from the gene expression data and the opposite. We evaluated our mapping method on a paired dataset with SNPs and gene expression data. The results gives 6% average error for the predicted expression. Evaluation of the overall system on two types of cancer shows that our approach enhances the accuracy by up to 7.6% over the baseline individual classifiers.

1 Introduction

A Single Nucleotide Polymorphism (SNP) is a DNA variation in which a single nucleotide in the genome differs between members of the same biological species or paired chromosomes. In normal persons, SNPs are most commonly found between genes in non-coding areas. When SNPs are found with high frequency within a gene or in a regulatory area near a gene, it can be used as a biological marker to detect diseases. Recent research has shown that SNPs can be used to detect genetic abnormalities in cancer tissues[18]. Moreover, variations in gene expression profiles

R. Mohamed(✉) · N.M. Ghanem · M.A. Ismail
Computer and Systems Engineering Department, Alexandria University, Alexandria, Egypt
e-mail: reham.m.samir@gmail.com

M.S. Mohamad et al. (eds.), *10th International Conference on PACBB*,
Advances in Intelligent Systems and Computing 477,
DOI: 10.1007/978-3-319-40126-3_9

among human individuals can be used to detect different diseases. Therefore, many studies have been proposed to study the relationship between gene expression data and SNPs[2, 3]. Recent studies [1] also showed that SNPs can be used to predict the gene expression by using a mapping approach, where a significant correlation can be detected between gene expression with the same genetic variations for e.g. SNPs minor allele count.

On the other hand, microRNAs (miRNAs) are short (1925 nucleotides) non-coding single-stranded RNA molecules, which are cleaved from 70-100 nucleotide miRNA precursors. miRNAs regulate gene expression either at the transcriptional or translational level, based on specific binding to the complementary sequence in the coding or non-coding region of mRNA transcripts [4].

Recently, it has been shown that SNPs, miRNAs and genes microarray datasets can be used successfully for cancer classification. In [16], SNPs genotype data was used to detect breast cancer subtype using support vector machine classifier. MiRNA profiles have been also used to discriminate tumors of breast[5], lung[5, 6], etc. Moreover, feature selection approaches and different classifiers for cancer detection using miRNA profiles were explored in [7]. While these approaches used the expression data for detecting cancer and its subtypes, they did not exploit the relationships between the different features (genes, SNPs and miRNAs). Few attempts have been made to exploit the relations between miRNA and mRNA expression profiles, such as: [8]. However, this approach assumes paired miRNA and mRNA data for each patient. Recently, a new approach was proposed in [12] which co-trains miRNA and gene expression classifiers by exploiting the correlation between miRNAs and genes.

The idea of co-training has been used in several fields [11, 19]. Co-training is a semi-supervised machine learning approach, where labeled and unlabeled datasets of different features are used to enhance the classification accuracy. Co-training can be useful for the purpose of microarray classification, where the data is usually provided for a small number of samples. This usually occurs because the collection of samples for microarrays is expensive and time consuming, therefore cannot cope with the large number of individuals to be tested. Co-training have proved to be useful for enhancing the accuracy of miRNA and gene expression classifiers. However, it has not been used for SNPs array data.

In this paper, we propose a new semi-supervised approach for detecting cancer and its subtypes using expression profiles data. We leverage the correlation between SNPs and gene profiles, and miRNA and gene profiles to co-train three parallel classifiers using unlabeled datasets. We build three individual SVM classifiers using labeled datasets for SNPs, gene and miRNA expressions, respectively. Then, we use an unlabeled gene expression dataset to co-train the SNPs and miRNA classifiers, and two unlabeled datasets for SNPs and miRNAs to co-train the gene classifier. We use the technique proposed in [12] to detect the relation between gene and miRNA expressions. For predicting the relations between SNPs and gene expressions, we propose a new technique that applies a KNN classifier on parallel datasets, that include the SNPs and gene expressions for the same individuals.

We tested the overall system on two different cancer types: breast cancer and prostate cancer. For breast cancer, our approach enhances the results by 7.6% over

the baseline SNP classifier and 2.8% over the baseline miRNA classifier. For prostate cancer, our approach enhances the results by 3.6% over the baseline SNP classifier and 3.1% over the baseline gene classifier.

The rest of the paper is organized as follows: Section 2 shows the related work to our system. Section 3 describes the system details. In Section 4, we show our experimental results. Finally, we conclude the paper in Section 5.

2 Related Work

Over the years, several research studies have been proposed to detect cancer status and subtypes using genomic data, such as: gene, miRNA expression profiles and SNPs genotypes. They have been used to detect different types of cancer, including breast cancer [5], lung [5, 6], etc. The previous papers used supervised machine learning techniques to detect cancer or classify it into one of its subtypes. SNPs microarray data have been also used for detection of breast cancer [17], prostate cancer [15], lung cancer [17], among others.

Several enhancements have been applied [7, 8, 14] to increase the accuracy of cancer detection, including applying feature selection methods, such as: Pearson and Spearman correlations, mutual information, etc. However, these approaches are all supervised, where they require labeled data to detect cancer. Enhancing the classification accuracy by building two classifiers using miRNA data and mRNA data was first explored in [8]. This approach applies combines the two bagged fuzzy KNN classifiers using fusion decision rule. However, the approach is limited as it assumes the existence of both miRNA and mRNA data for each patient, which is hard and expensive to achieve on a large scale. The use of unlabeled data was explored in [12]. This approach uses semi-supervised learning to detect cancer subtypes. They use the publicly available unlabeled sets to enrich the training data of the classifiers using self-learning and co-training to combine both miRNA and gene expression sets. Other techniques were proposed that perform mapping between different biological features, as [9, 10].

In this paper, we extend the previous work by integrating the gene expression, miRNA expression and SNPs microarrays to co-train three classifiers for detecting cancer and its subtypes. To the best of our knowledge, this work is the first to apply a semi-supervised machine learning technique on the three biological features for cancer detection. We also propose a new technique to predict the relation between gene expression and SNPs microarray data by applying KNN on a parallel dataset. Our technique is discussed in more details in the next section.

3 Co-Training of SNP/Gene/MiRNA Based Classification

Figure 1 shows our system architecture. The system consists of three individual classifiers: SNPs, gene and miRNA expression classifiers. Each classifier is trained using a separate labeled dataset. Then, an unlabeled dataset for each classifier is used

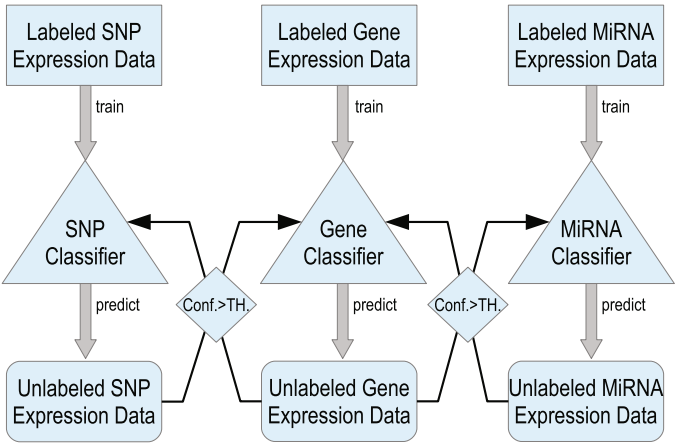


Fig. 1 System architecture.

to co-train another classifier. Each instance in the unlabeled dataset is predicted using its classifier. If the confidence ($Conf.$) of the prediction result is above a threshold ($TH.$), this instance with the predicted class label is used to train the other classifier. The gene unlabeled dataset is used to co-train the SNPs and miRNA classifiers respectively, while each of the SNP and MiRNA unlabeled datasets is used to co-train the gene classifier only. This approach not only enhances the accuracy of each classifier, but also provides insight to increase the available microarray datasets that could be used for different purposes.

In the next subsections, we show the co-training details of each classifier type.

3.1 Co-Training of MiRNA/Gene Classifier

Previous studies have shown that there are many-to-many relations between genes and miRNAs. We exploit these relations to predict the miRNA expression values from gene expression values and vice versa. We use miRanda [13] database which provides a set of genes and their related miRNAs. The approach of co-training the miRNA classifier can be summarized in the following points:

1. MiRNA and gene classifiers are built separately using labeled datasets, which are labeled manually with cancer status (normal or tumor) or subtype.
2. Each instance of the gene unlabeled dataset is fed into the gene classifier to predict it class label.
3. If the prediction confidence is above a threshold $TH.$, then this instance is used to co-train the miRNA classifier.

4. The gene expression instance is mapped into a miRNA expression instance, where the expression value of each miRNA is equal to the average of the expression values of the related genes.
5. Finally, the mapped miRNA expression instance is fed into the miRNA classifier to be used as a training instance.

3.2 Co-Training of SNP/Gene Classifier

In [1], SNPs genotype data has been used to predict gene expression data. In this paper, we use the SNP array expression data to predict the gene expression data of the same individual. Given two parallel datasets, which provide the SNPs array data and the gene expression data for 83 individuals, we use a K-Nearest Neighbor (KNN) classifier to predict the gene expression of a new individual given his SNPs expression. We calculate the Euclidean distance between the test SNP expression sample and the SNP training dataset. The predicted gene expression is the average of the gene expressions of the K nearest SNP expressions. The co-training approach can be summarized as follows:

1. The gene and SNPs classifiers are constructed using their labeled datasets.
2. The class label of each unlabeled gene expression instance is predicted using the gene classifier.
3. Instances with confidence above a threshold are used for co-training.
4. The SNPs expression of these instances is predicted using KNN. First, we calculate the nearest K gene expression to the new instance. Then, we take the average of the SNPs expression of the K neighbors as the predicted SNPs expression.
5. The predicted SNPs expression is used to co-train the SNPs classifier.

3.3 Two-phase Co-Training

The gene classifier is co-trained using two phases. First, it takes the instances from the miRNA unlabeled data which are classified using the miRNA classifier. The instances classified with confidence larger than a threshold ($TH.$) are mapped to gene expression, and are added to the gene classifier training data. Similarly, the SNPs unlabeled data, which are classified with high confidence, are mapped to gene expression using KNN as discussed before. The mapped instances are added to the gene classifier training data. Final, the gene classifier is trained using the integrated dataset, which includes the original gene expression training data, the instances mapped from the miRNA and SNPs unlabeled datasets.

4 Experimental Results

4.1 Datasets

The system is evaluated using different cancer datasets from GEO[20]. We apply two types of experiments to evaluate our approach. First, we evaluate our SNP/Gene expression mapping methodology using an aligned dataset which contains the SNP and Gene expressions for the same individuals. Second, we evaluate the overall system using breast cancer and prostate cancer datasets. For each cancer type, we train the classifiers using a labeled dataset, then we use the trained classifiers to predict the labels of unlabeled datasets. The unlabeled datasets of each classifier with the predicted labels are then used to train the other classifiers.

For the breast cancer datasets, the data is labeled with the state of the ER/HER2 receptors, which infer the breast cancer subtype. Each receptor takes a positive or negative value, leading to four cancer subtypes. Each instance is classified to one of the four states. For prostate cancer, the datasets are labeled with the tissues state as normal or tumor. In this case, the task of the classifier is to predict the state of each new instance as normal or tumor.

4.2 SNP/Gene Expression Mapping Evaluation

We evaluated our mapping method using GSE33356 dataset [21], which contains both, SNPs microarray and gene expression data for 84 individuals. We divided the data into two equal sets for training and testing. For each individual in the test data, we predict the gene expression using our KNN classifier and compare it with the original gene expression of this individual, by calculating the average distance over all the test set data averaging over all genes in the expression. We do the same for the SNPs microarrays. The results give an average distance of 0.39 between the predicted and the original gene expressions with 95% confidence interval of [0.37, 0.42]. This is equivalent to 6% error for the average gene expression where the average value of the gene expression is 6.4. For SNPs microarrays the results give 0.34 average distance with [0.32, 0.36] as the 95% confidence interval.

4.3 Overall System Evaluation

In this section, we show our experimental results over two datasets: Breast cancer and prostate cancer. For breast cancer, the data is labeled with the cancer subtype. For prostate cancer, the data is labeled with the cancer status as: normal or tumor. We used Weka implementation[1] to build our classifiers. Each individual classifier is an SVM classifier. In the following subsections, we show the results of our system and compare them to the baseline individual SVM classifiers.

[1] http://www.cs.waikato.ac.nz/ml/weka/

Table 1 Gene expression labeled data distribution, E=ER and H2=HER2.

Type	Breast Cancer				Prostate Cancer	
Sub-type	E+/ H2-	E-/ H2+	E-/ H2-	E+/ H2+	N	T
Train	17	9	14	4	20	20
Test	17	8	14	4	20	20

Table 2 SNPs expression labeled data distribution, where E=ER and H2=HER2.

Type	Breast Cancer				Prostate Cancer	
Sub-type	E+/ H2-	E-/ H2+	E-/ H2-	E+/ H2+	N	T
Train	8	5	9	4	28	40
Test	7	3	8	4	27	39

Table 3 MiRNA expression labeled data distribution.

Type	Breast Cancer			
Subtype	ER+/ Her2-	ER-/ Her2+	ER-/ Her2-	ER+/ Her2+
Train	23	5	14	4
Test	28	5	10	3

Tables 1, 2 and 3 show the distribution of the classes of the training and testing datasets. The size of unlabeled datasets is shown in Table 4.

4.4 Breast Cancer

Breast cancer is one of the most common types of cancer. Previous studies have shown that the ER and PR statuses can be used to detect the cancer subtype. For the SNPs classifier, we use GSE26232 as the SNPs labeled dataset and GSE37035 as the unlabeled dataset. For the gene classifier, we use GSE20713 as the labeled dataset and GSE22865, GSE16179 and GSE32161 as the unlabeled dataset. Finally, we use GSE19536 for the labeled dataset of miMRNA and GSE26659 for the unlabeled dataset. Cross validation and test results are shown in Table 5. The results show that co-training of the SNPs classifier improves the results by 7.6% in the F-measure. While the co-training of the miRNA classifier improves its results by 2.8%.

4.5 Prostate Cancer

We test our approach on prostate cancer data labeled with the cancer status (Normal or Tumor). For the SNPs classifier, we use GSE18333 and GSE29569 as the SNPs labeled dataset and GSE27105 as the unlabeled dataset. For the gene classifier, we use GSE32448 as the labeled dataset and GSE37199 as the unlabeled dataset. Results are shown in Table 5. The results show that co-training of the SNPs classifier improves

Table 4 Unlabeled datasets

	Sample Size	miRNA/gene/ SNPs Number	Title
Breast Cancer (GSE37035)	66	909622 SNPs	Polarity gene alterations in pure invasive micropapillary carcinomas of the breast.
Breast Cancer (GSE22865)	12	54675 genes	mRNA expression is a strong prognostic biomarker in breast and ovarian cancer.
Breast Cancer (GSE16179)	19	54675 genes	BT474 and BT474-J4 microarray data.
Breast Cancer (GSE32161)	6	54675 genes	Microarray analysis of genes associated with cell surface NIS protein levels in breast cancer.
Breast Cancer (GSE26659)	94	147 miRNA	microRNA and cancer progression in breast tumors.
Prostate Cancer (GSE27105)	50	909622 SNPs	Genomic Signatures of Metastasis in Prostate Cancer.
Prostate Cancer (GSE37199)	60	54675 genes	Blood mRNA expression signatures derived from unsupervised analyses identify prostate cancers with poor outcome.

Table 5 Results

	Breast Cancer						Prostate Cancer					
	Cross Validation			Test			Cross Validation			Test		
SNPs classifier	93.7	92.3	93	72.7	63.6	67.8	92.8	92.6	92.7	52.5	50	51.2
Co-trained SNPs	N/A	N/A	N/A	71.2	77.3	74.1	N/A	N/A	N/A	55	54.5	54.8
Gene classifier	60.3	52.2	56	65.6	67.4	66.5	80	80	80	73	72.5	72.7
Co-trained gene	N/A	N/A	N/A	61.4	60.5	60.9	N/A	N/A	N/A	75.2	75	75.1
miRNA classifier	57.1	63	59.9	64.2	63	63.6	–	–	–	–	–	–
Co-trained miRNA	N/A	N/A	N/A	70.1	63	66.4	–	–	–	–	–	–

the results by 3.6% in the F-measure. While the co-training of the gene classifier improves the F-measure by 3.1%.

5 Conclusion

In this paper, we show a new semi-supervised system for detecting cancer and its subtypes. We exploit the relations between SNPs, gene and miRNA expression data to co-train each classifier using mapped data from the other classifier, to enrich the dataset of each classifier. We test the system on two datasets. Our results showed that the co-training approach enhanced the classification accuracy.

References

1. Manor, O., Segal, E.: Robust prediction of expression differences among human individuals using only genotype information. PLoS Genet. (2013)
2. Stranger, B.E., et al.: Relative impact of nucleotide and copy number variation on gene expression phenotypes. Science **315**(5813) (2007)
3. Dixon, A.L., et al.: A genome-wide association study of global gene expression. Nature Genetics **39**(10) (2007)
4. Katayama, Y., et al.: Identification of pathogenesis-related microRNAs in hepatocellular carcinoma by expression profiling. Oncology Letters **4**(4) (2012)
5. Volinia, S., et al.: A microRNA expression signature of human solid tumors defines cancer gene targets. Proceedings of National Academy of Sciences, USA (2006)
6. Yanaihara, N., et al.: Unique microRNA molecular profiles in lung cancer diagnosis and prognosis. Cancer Cell **9**(3) (2006)
7. Kim, K., Cho, S.: Exploring features and classifiers to classify microRNA expression profiles of human cancer. Neural Information Processing (2010)
8. Wang, Y., et al.: Classifier fusion for poorly-differentiated tumor classification using both messenger RNA and microRNA expression profiles. In: Computational Systems Bioinformatics Conference (2006)
9. Manor, O., Segal, E.: GenoExp: a web tool for predicting gene expression levels from single nucleotide polymorphisms. Bioinformatics (2015)
10. Cheng, H., et al.: Fine mapping of QTL and genomic prediction using allele-specific expression SNPs demonstrates that the complex trait of genetic resistance to Mareks disease is predominantly determined by transcriptional regulation. BMC (2015)
11. Mihalcea, R.: Co-training and Self-training for Word Sense Disambiguation. CoNLL (2004)
12. Ibrahim, R., et al.: miRNA and gene expression based cancer classification using self-learning and co-training approaches. Bioinformatics and Biomedicine (2013)
13. John, B., et al.: Human microRNA targets. PLoS Biol. **2**(11) (2004)
14. Zheng, Y., Kwoh, C.: Cancer classification with microRNA expression patterns found by an information theory approach. Journal of Computers **1**(5) (2006)
15. Dumur, C.I., et al.: Genome-wide detection of LOH in prostate cancer using human SNP microarray technology. Genomics **81**(3) (2003)
16. Upstill-Goddard, R., et al.: Support vector machine classifier for estrogen receptor positive and negative early-onset breast cancer. PloS One **8**(7) (2013)
17. Zhao, X., et al.: An integrated view of copy number and allelic alterations in the cancer genome using single nucleotide polymorphism arrays. Cancer Research (2004)
18. Dutt, A., Rameen, B.: Single nucleotide polymorphism array analysis of cancer. Current Opinion in Oncology **19**(1) (2007)
19. Ibrahim, R., et al.: Context-aware semi-supervised motif detection approach. Engineering in Medicine and Biology Society (2014)
20. Edgar, R., Domrachev, M., Lash, A.E.: Gene Expression Omnibus: NCBI gene expression and hybridization array data repository. Nucleic Acids Res. (2002)
21. Lu, T.P., Lai, L.C., Tsai, M.H., Chen, P.C. et al.: Integrated analyses of copy number variations and gene expression in lung adenocarcinoma. PLoS One (2011)

Systematic Evaluation of Gene Expression Data Analysis Methods Using Benchmark Data

Henry Yang

Abstract Due to limited amount of experimental validation datasets, data analysis methods for identifying differential expression based on high-throughput expression profiling technologies such as microarray and RNA-seq cannot be statistically validated properly, and thus guidelines for selecting an appropriate method are lacking. We applied mRNA spike-in approaches to develop a comprehensive set of experimental benchmark data and used it to evaluate various methods for identification of differential expression. Our results show that using the median log ratio to identify differential expression is superior to more complex and popular methods such as modified *t*-statistics. The median log ratio method is robust that a reasonably high accuracy of identification of differentially expressed genes can be achieved even for data with a small number of replicates and strong experimental variation between replicates. Machine learning for classification of differential expression based on the benchmark dataset indicates the existence of even more accurate methods for identification of differential expression. With this dataset, it can be also demonstrated that the methods prediction of false discovery rate based on a small number of replicates could be very inaccurate.

Keywords Differential expression · Statistical method evaluation · Spike-in validation

1 Introduction

In comparative studies, expression profiles between two states (phenotypes) can uncover genes that are differentially expressed providing information relating the

H. Yang(✉)
Cancer Science Institute of Singapore, National University of Singapore,
14 Medical Drive #12-01, Singapore 117599, Singapore
e-mail: csiyangh@nus.edu.sg

M.S. Mohamad et al. (eds.), *10th International Conference on PACBB*,
Advances in Intelligent Systems and Computing 477,
DOI: 10.1007/978-3-319-40126-3_10

phenotypic changes to the genotypic ones. With the development of microarray and latterly next-generation sequencing (NGS), the whole transcriptome can be easily profiled. A problem associated with gene expression profiling using microarray or NGS (RNA-seq) is however a limit in the number of replicates due to e.g. sample limitation and/or cost effectiveness. As such, statistical data analysis for differential expression (DE) is often carried out with a small number of sample replicates. However, most statistical methods are more suited only for a larger number of replicates. Thus it is important to evaluate the performance of those statistical methods for a small number of replicates with potential large variation.

There are numerous techniques available for DE identification. The methods can be broadly categorized into those that are applied to a single array and those that are applied to a set of replicates. The simplest single-array methods for DE identification are minimal fold change and percentile cutoff. In these methods, the criteria for determining DE are given by using a fixed cutoff value for minimal fold change or percentile, and self-hybridization comparison can be used to obtain the appropriate cutoff value (LERSD) (1). Many DE identification methods require replicates. They are based on e.g. *t*-statistic, ANOVA, B statistic, hierarchical modeling (2), Wilcoxon rank sum statistic, and modified *t*-statistic including Significance Analysis of Microarrays (SAM) (3) and modified SAM with ranking (*samroc*) (4). Most of these statistical methods were developed more than a decade ago. Recently, only a few methods have been proposed. They are however in fact a hybrid of single-array methods using fold change and the above statistical methods (*p* value or variance) (5,6), without providing any novel statistical methodology.

In order to select an appropriate method for DE identification, experimental validation of whole transcriptomic data obtained from microarray or RNA-seq is critical. Such experimental validation datasets can be used not only for evaluation of existing analysis methods but also for initiating further development of statistical analysis methods for more accurate DE identification. Traditional validation of expression results is done using quantitative RT-PCR/qPCR or Northern blot. These procedures are often tedious. Furthermore, they are also expression detection platforms and may yield more accurate expression intensities and thus more reliable fold changes, but they still cannot provide an accurate answer to which genes are differentially expressed and which genes are not. Recently, TaqMan qPCR data of 1000 genes were used to evaluation of DE analysis methods for RNA-seq data (7), but the "true" positives ("truly" differentially expressed genes) were called on the basis of the qPCR data. Spiking of mRNAs is an effective alternative as spike-in genes readily constitute the true positives. To be useful for statistical analysis, the number of spike-in genes should be sufficiently large. However, spiking of a large number of genes has not been performed so far in the literature. In this work, we have developed an experimental benchmark dataset with a large number of spike-in genes for microarrays (which can be also extended to NGS data) and used it to validate various DE identification methods.

The dataset was established by spiking of 169 genes at various concentrations with multiple replicates. We evaluated various DE identification methods using this benchmark dataset and found the median log ratio to be the most accurate and robust method. Furthermore, we showed that current approaches to predict false discovery rate (FDR) severely underestimated the true FDR. With this benchmark dataset, we also demonstrated that novel statistical methods could be further developed for more accurate DE identification.

2 Material & Methods

2.1 Spike-in Experiment

The spike-in experiment was used for generating validation datasets by spiking 169 transcripts into mRNA isolated from mouse hybridoma CRL1606 cells at 11 different concentrations (ranging from 0-2 pmol for each gene) with three array replicates for each of: 0.025, 0.15, 0.2, 1 and 2 pmol, and six replicates for each of: 0, 0.05, 0.1, 0.25, 0.5, and 0.75 pmol. The microarrays used for hybridizations are mouse arrays containing ~7.8k clones and each clone is spotted on arrays in duplicates. Detailed description of the microarrays can be found in a previous work (8). The spotted duplicates for each gene were treated as two independent spot replicates, resulting in 6 or 12 spot replicates in total.

2.2 Validation Strategy

The benchmark data provides a set of truly differentially expressed genes (true positives). As different DE identification methods yield different sets of predicted differentially expressed genes, it is possible to compare the prediction errors generated by various techniques and evaluate their effectiveness. A popular approach for estimation of prediction errors is the receiver operating characteristics (ROC) curve. To obtain a ROC curve, the FDR versus the True Discovery Rate (TDR) is plotted.

2.3 Preprocessing & Normalization

The intensities of all spots were background subtracted. If the intensity after subtracting the background is smaller than twice the standard deviation of the overall background, the intensity is assigned to twice this standard deviation (9). As the proportion of differentially expressed genes is small compared to the total number of genes (clones), normalization was performed with the global Cross-Correlation method (8). Prior to DE identification, each array was preprocessed and normalized according to the above methods.

3 Results

3.1 Comparison of Various DE Identification Methods

Figure 1 compares the performance of several DE identification methods: LERSD (1), Variance-modeled posterior inference (Vampire) (2), *t*-statistic, SAM (3), *samroc* (4), and median log ratio (MDLR). The MDLR is obtained by evaluating the median of the log ratios over the replicates for a gene and comparing it against a fixed threshold. It can be seen from Figure 1 that MDLR consistently outperforms the counterparts, whilst Vampire and SAM/samroc perform differently with different sets of replicates (at different spike-in concentrations).

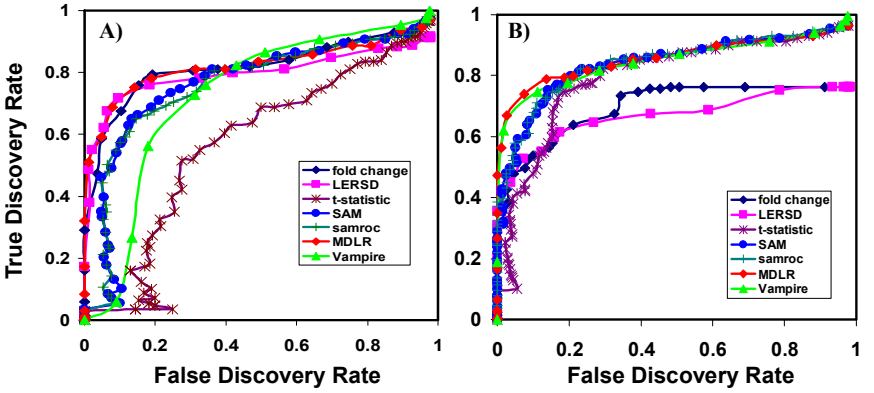


Fig. 1 Comparison of different DE identification methods for two different sets of replicates at two different spike-in concentrations: (A) 0.2pmol with 6 replicates & (B) 0.25pmol with 12 replicates

3.2 Effects of Replicate Number & Strong Experimental Variation

With an increasing number of replicates, Figure 2B,C show a steady improvement in the performance of the *t*-statistic or modified *t*-statistic. On the contrary, the performance of MDLR does not vary much as the number of replicates increases (Fig 2D), implying consistently good performance of MDLR even for small sample sizes. Similar results can be found for other sets of replicates at other spike-in concentrations.

3.3 Existence of More Accurate DE Identification Methods

A Supporting Vector Machine (SVM) classifier was used to determine whether there exists another statistic, say hypothetical *s*-statistic: $s=f(MD,\sigma_M)$, which can

predict differentially expressed genes more accurately than MDLR, where *MD* and σ_M are the respective median and standard deviation cross the replicates of a gene. After training the SVM classifier using the spike-in microarray data over all genes and all spike-in concentrations, a single solution produced the maximum TDR value at FDR=0. This solution gives a better prediction than MDLR across different spike-in concentrations (Fig 3). Although this optimal solution cannot be explicitly formulated, it indicates the existence of more appropriate statistical methods for DE identification than MDLR. Figure 3 also shows that the inclusion of σ_M improves the prediction power of the *s*-statistic, particularly for replicates with strong variation (e.g. at the spike-in concentration of 0.75 pmol). This seems contradictory to the finding that MDLR is superior to the σ_M incorporating *t*-statistic and its variants (Figs 1-2). The possible explanation is that the standard deviation (σ_M) is relevant for accurate DE identification, particularly for array replicates with strong variation. However, current statistical methods are unable to effectively take advantage of the inclusion of σ_M.

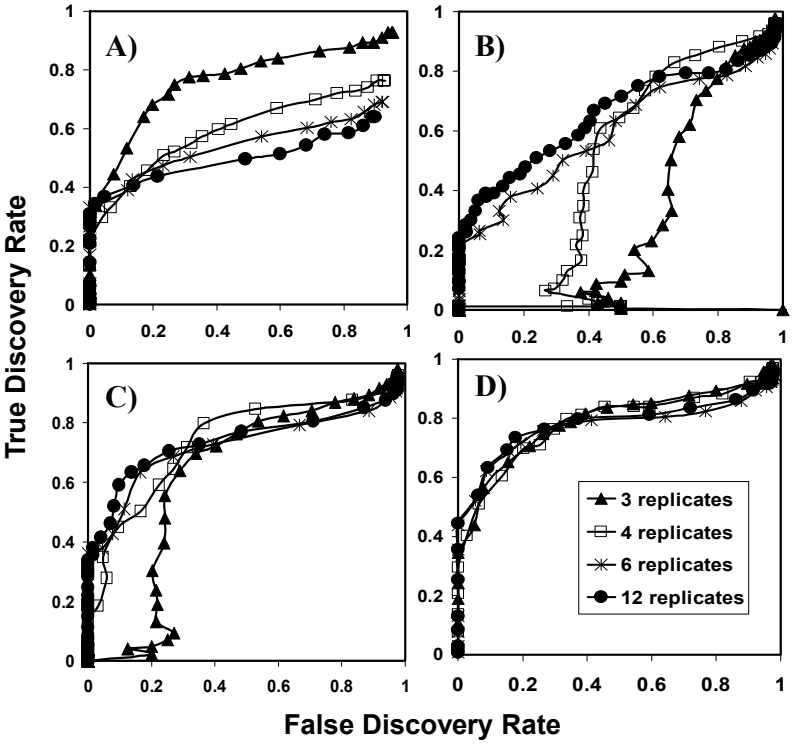


Fig. 2 Comparison of performance of four DE identification methods with an increasing number of replicates at the spike-in concentration of 0.1 pmol: A) fold change, B) *t*-statistic, C) SAM, and D) MDLR.

3.4 *Accuracy in DE Identification & FDR Prediction Using Permutation*

Based on the Affymetrix Hu95 spike-in dataset, which contains only 14 spike-in genes, a large number of differentially expressed genes were generated using the bootstrapping-based permutation, enabling statistical evaluation of analysis methods (4). In order to evaluate the accuracy of such permutation, we randomly selected 16 spike-in genes and then used bootstrapping to generate 169 differentially expressed genes as permutated true positives. In Figure 4A&B, the bootstrapping results were compared with the original ones. Apparent discrepancies between the original and permutated results indicate that current methods of permutation are not able to accurately mimic the results from actual experimental data.

The prediction power of a DE identification method is dependent on a control parameter. SAM is a popular DE identification method as it not only provides a statistic to identify differentially expressed genes but also predicts the FDR at a given control parameter enabling the user to control the FDR at a suitable level. However, the accuracy of the FDR prediction using SAM or other permutation techniques has never been evaluated or questioned for gene expression data with a small number of replicates. Upon comparing the predicted FDR with the actual FDR, we found that SAM considerably underestimates the FDR (Fig 4C&D). Most FDR prediction procedures consist of two steps: estimation of π_0 (fraction of non-differentially expressed genes) and subsequent prediction of FDR. To check the accuracy of the latter step, we provided the correct value for π_0. Figure 4C&D show that the FDR is still grossly underestimated (Fig 4C&D) even when providing the correct π_0.

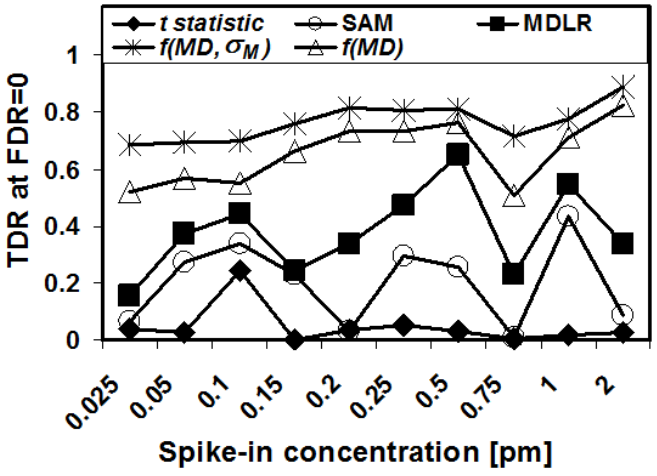


Fig. 3 TDR obtained at FDR=0 across different spike-in concentrations using different DE identification methods including the hypothetical *s*-statistics with/without σ_M.

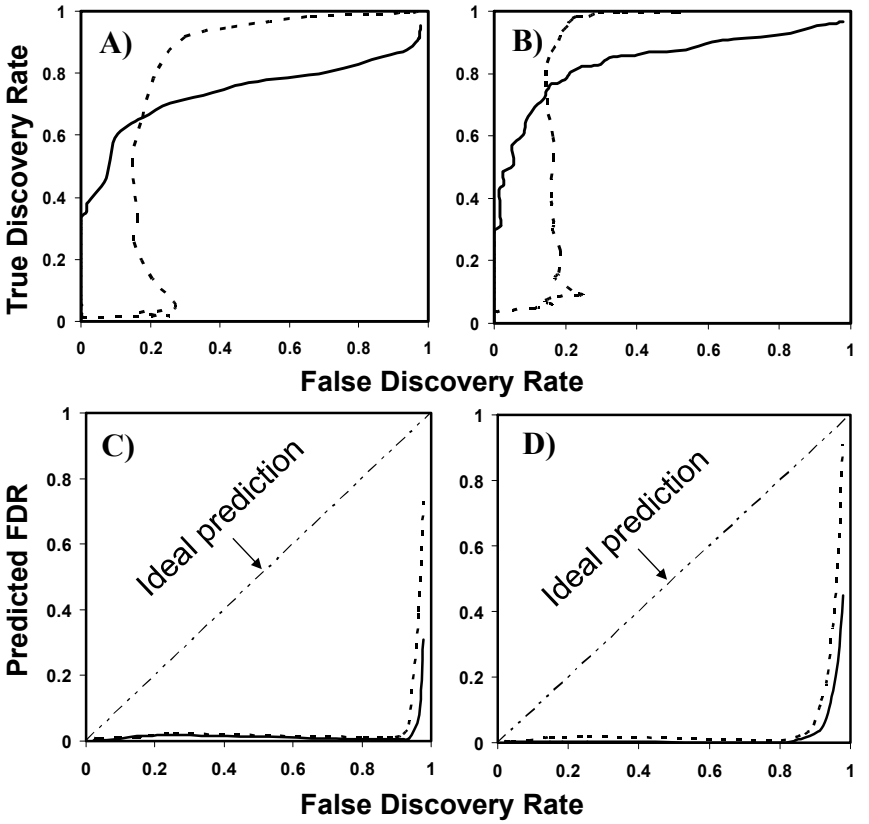


Fig. 4 Comparison of permutated results with the original results for two sets of replicates at two different spike-in concentrations (0.1pmol in A & C and 0.25pmol in B & D) using all available replicates. A & B show ROC curves based on spike-in genes (solid lines) and bootstrapping to increase the number of spike-in genes (dashed lines). C & D show FDR prediction using permutation described in SAM (solid lines: predicted π_0; dashed lines: true π_0).

4 Conclusions

Our spike-in experiments surprisingly showed that elimination of the standard deviation results in a more accurate and robust DE identification technique. The robustness of MDLR has been proven by its consistent performance both over a varying number of replicates and with replicates of strong experimental variation. Furthermore, our approach using a SVM classifier not only indicates the existence of novel statistical methods that can outperform MDLR, but also shows the need for inclusion of the standard deviation. This implies that the statistical methods in current formulations have not incorporated the standard deviation appropriately, or the estimation of standard deviation using a small number of replicates is inappropriate.

It should be noted that our approach outlined in this paper can be also utilized for evaluation of DE identification using RNA-seq data. This study not only evaluates the performance of various DE identification techniques but also highlights the importance of validating these theoretically based techniques experimentally. Many highly regarded DE identification statistical methods with sound theoretical grounding have been shown to be ineffective when actual experimental data is presented. This is largely due to the inability of current statistical methods to handle data with small sample sizes and large experimental variation between replicates. As such, using a simple statistic – MDLR yields better results than all of the more sophisticated and widely accepted statistical methods.

References

1. Yang, H., Haddad, H., Tomas, C., Alsaker, K., Papoutsakis, E.T.: A segmental nearest neighbor normalization and gene identification method gives superior results for DNA-array analysis. Proc. Natl. Acad. Sci. USA **100**, 1122–1127 (2003)
2. Hsiao, A., Worrall, D.S., Olefsky, J.M., Subramaniam, S.: Variance-modeled posterior inference of microarray data: detecting gene-expression changes in 3T3-L1 adipocytes. Bioinformatics **20**, 3108–3127 (2004)
3. Tusher, V.G., Tibshirani, R., Chu, G.: Significance analysis of microarrays applied to the ionizing radiation response. Proc. Natl. Acad. Sci. USA **98**, 5116–5121 (2001)
4. Broberg, P.: Statistical methods for ranking differentially expressed genes. Genome Biol. **4**, R41 (2003)
5. Mazurek, U., Owczarek, A., Nowakowska-Zajdel, E., Wierzgon, J., Grochowska-Niedworok, E., Kokot, T., Muc-Wierzgon, M.: Statistical analysis of differential gene expression in colorectal cancer using CLEAR-test. J. Biol. Regul Homeost. Agents **25**, 279–283 (2011)
6. Vaes, E., Khan, M., Mombaerts, P.: Statistical analysis of differential gene expression relative to a fold change threshold on NanoString data of mouse odorant receptor genes. BMC Bioinformatics **15**, 39 (2014)
7. Rapaport, F., Khanin, R., Liang, Y., Pirun, M., Krek, A., Zumbo, P., Mason, C.E., Socci, N.D., Betel, D.: Comprehensive evaluation of differential gene expression analysis methods for RNA-seq data. Genome Biol. **14**, R95 (2013)
8. Chua, S.W., Vijayakumar, P., Nissom, P., Yam, C.Y., Wong, V.T., Yang, H.: A novel normalization method for effective removal of systematic variation in microarray data. Nuclei Acid Research **34**, e38 (2006)
9. Tomas, C., Alsaker, K., Bonarius, H., Hendriksen, W., Yang, H., Beamish, J.A., Paredes, C., Papoutsakis, E.T.: DNA array-based transcriptional analysis of asporogenous, nonsolventogenic Clostridium acetobutylicum strains SKO1 and M5. J. Bacteriol. **185**, 4539–4547 (2003)

A Clustering-Based Method for Gene Selection to Classify Tissue Samples in Lung Cancer

José A. Castellanos-Garzón, Juan Ramos, Alfonso González-Briones and Juan F. de Paz

Abstract This paper proposes a gene selection approach based on clustering of DNA-microarray data. The proposal has been aimed at finding a boundary gene subset coming from gene groupings imposed by a clustering method applied to the case study: gene expression data in lung cancer. Thus, we assume that such a found gene subset represents informative genes, which can be used to train a classifier by learning tumor tissue samples. To do this, we compare the results of several methods of hierarchical clustering to select the best one and then choose the most suitable clustering based on visualization techniques. The latter is used to compute its boundary genes. The results achieved from the case study have shown the reliability of this approach.

Keywords DNA-microarray · Feature selection · Data clustering · Genetic algorithm · Data mining · Visual analytics

1 Introduction

Lung cancer is one of the most common types of malignancies worldwide and one of the most frequent causes of death in developed countries, constituting 27% of all cancer deaths. Thus, early diagnosis is essential for the patient's survival. Unfortunately, most patients are diagnosed at an advanced stage of the disease, in which they have already developed metastases [1]. Such an event is a consequence of the lack of early symptoms, which do not appear until the disease is in a critical condition. Hence, this proliferative syndrome presents a high risk of metastasis which binds to the absence of effective treatments. This has led researchers to develop classifiers based

J.A. Castellanos-Garzón(✉) · J. Ramos · A. González-Briones · J.F. de Paz
Faculty of Science, Biomedical Research Institute of Salamanca/BISITE Research Group,
Edificio I+D+i USAL, University of Salamanca, c/ Espejo s/n, 37007 Salamanca, Spain
e-mail: {jantonio,juanrg,alfonsogb,fcofds}@usal.es

M.S. Mohamad et al. (eds.), *10th International Conference on PACBB*,
Advances in Intelligent Systems and Computing 477,
DOI: 10.1007/978-3-319-40126-3_11

on microarray technology and able to support metastasis diagnosis and prognosis toward different organs [2].

The study of gene expression data from DNA-microarrays is of great interest for Bioinformatics (and functional genomics), because it allows us to analyze expression levels in hundreds of thousands of genes in a living organism sample. This feature makes gene expression analysis a fundamental tool of research for human health [3]. It provides identification of new genes being key in the genesis and development of diseases. However, the exploration of these large datasets looking for a small subset of significative genes is a crucial but very difficult issue. The use of data mining techniques along with information visualization technology can help to cope with this problem by improving the data analysis process [4].

In this context, feature/gene selection involves an important research topic in gene expression data, leading to the gene discovery relevant for a particular target annotation. Such genes are called *informative genes* or *differentially expressed genes*, since they are able to differentiate samples from different populations [5]. According to that, this paper presents a gene selection method for classification, where statistical significance tests (p-value, variance), clustering and boundary gene selection have been used to discover an informative gene subset from a DNA-microarray study in lung cancer.

2 A Gene Selection Approach for Classification

This section explains each component of our gene selection approach, which has two filtering components before applying a clustering method. Once clustered the data, the boundary genes of the clustering are computed as the end step. Boundary genes are data points that are located at the margin of densely distributed data, and are very useful in data mining applications, representing a subset of the population that possibly belongs to two or more classes [6]. Awareness of these points is also useful in classification tasks, since they can potentially be misclassified [7]. In consequence, boundary points are good candidates to be informative genes.

2.1 Developing the Feature Selection Model

As previously said, the model pursued by our approach consists of three filtering processes and a cluster analysis process as shown in Figure 1, all aimed at finding a significative gene subset, i.e., informative genes. So the strategy followed by our proposal to reach a gene subset can be described into four stages according to Figure 1:

1. *Stage FP-I:* This is the initial stage applied to the target dataset. This stage is responsible for carrying out a significance test by relating genes to the studied disease, in this case, tumor tissue samples. Therefore, Mann-Whitney test has

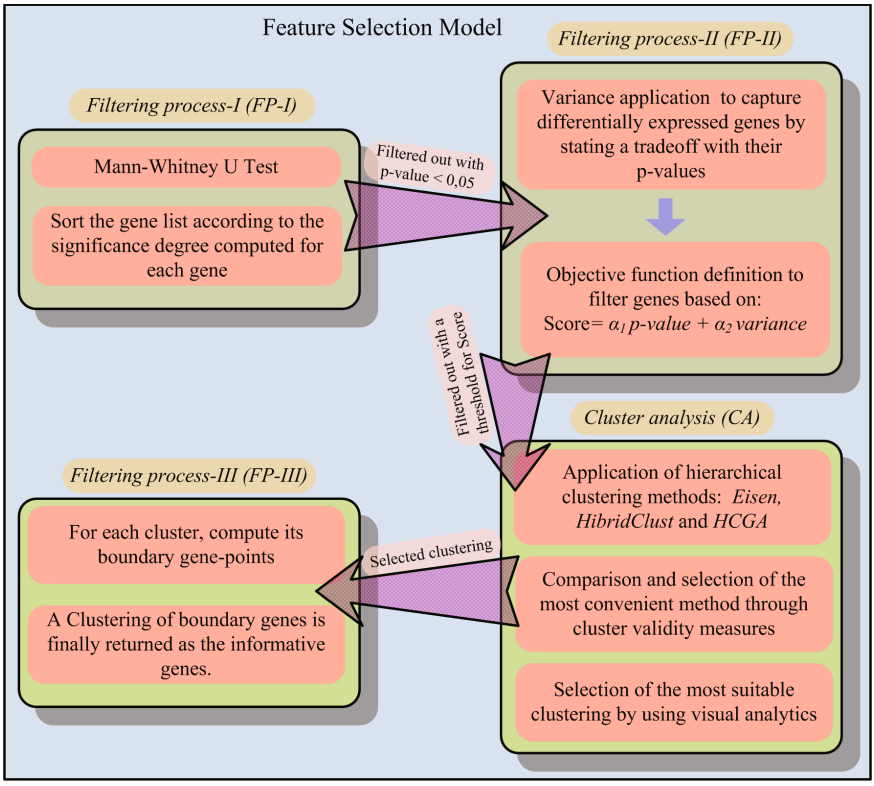


Fig. 1 Steps followed by the gene selection method. There are three filtering steps, FP-I, II and III, plus a core process of cluster analysis (CA), acting as a link between the filtering processes.

been chosen for that end. Mann-Whitney test is a nonparametric one stating a null hypothesis relating samples to the same population whereas the alternative hypothesis relates samples to different populations [8]. Thus, once applied this test, genes with *p-value* under 0.05 are filtered out towards the next stage. Note that such genes are who reject the null hypothesis and in consequence, they have the greatest statistical significance.

2. *Stage FP-II:* This stage has as input a reduced and ordered dataset (genes with p-values < 0.05) coming from FP-I. This stage is intended to capture genes whose variation of their expression levels is meaningful with respect to the rest, whereas high significance is also kept. So those genes will present a higher variation on their expression levels at the same time that they are meaningful for the context. Hence, we combine the *variance* indicator to measure such variations with the p-value assigned to genes into an objective function in order to filter out those relevant genes. Then, the score given to a gene g holds the following objective function:

$$Score(g) := \alpha_1 \cdot pvalue(g) + \alpha_2 \cdot variance(g), \quad (1)$$

where α_1, α_2 are scalars which can be defined as $\alpha_1 = -1$ since it is in interval $[0, 1]$ and $\alpha_2 = \frac{1}{maxvar}$. $maxvar$ is the maximum gene variance in the dataset. In consequence, the larger the values of function Score the higher the gene relevance. This means finding small values for $pvalue$ against big values for $variance$ as a maximization process. Then, by assigning a score to each gene based on this function and defining a threshold to filter out those genes with high score, we achieve the result-dataset of this stage.

3. *Stage CA:* This stage has as input a dataset filtered by FP-II and is responsible for choosing a clustering favorable for the next process. To do this, three hierarchical clustering methods have been selected since they are used frequently in cluster analysis of DNA-microarray data. Therefore, the strategy is to compare their results on the current dataset, based on cluster validity measures, i.e., *homogeneity* and *separation*. The method with the best validity score is selected to be analyzed visually and then choose the suitable clustering.

   – *clustering methods:* the *Eisen* clustering method carries out an agglomerative hierarchical clustering in which each cluster is represented by the mean vector for data in the cluster [9]. Furthermore, this method has been one of the first methods bringing a visualization coupling heatmap with dendrogram. The $HybridHclust$ method is a divisive hierarchical clustering, which is applied to the data with constraint that mutual clusters cannot be divided. Within each mutual cluster, the divisive strategy is re-applied to yield a top-down hybrid in which mutual cluster structure is retained [10]. Meanwhile, *HCGA* is an agglomerative hierarchical clustering based on *genetic algorithms* as the search method. Hence, it uses the evolutionary force to alter and recombine dendrograms from generation to generation by achieving the most favorable ones [11].

4. *Stage FP-III:* In this stage the clustering selected from CA is processed to compute the boundary genes for each of its clusters. Then, after applying the previous filtering processes and since boundary genes are representative of each cluster imposed on the current dataset [4], the resulting boundary genes can discriminate the remaining genes and so they can be considered informative genes. We have used the *ClusterBoundary* algorithm to compute boundary genes in each cluster as defined in [4].

3 Results on the Case Study

This section outlines the application of our approach to a public dataset of lung cancer (repository NCBI, http://www.ncbi.nlm.nih.gov/sites/GDSbrowser?acc=GDS3627), which comes from DNA-microarray technology, being Affymetrix Human Genome U133 Plus 2.0 Array. This dataset discloses a comparison study of two non-small cell

lung cancer histological subtypes: adenocarcinomas (AC) and squamous cell carcinomas (SCC). The results provide insight into the molecular differences between AC and SCC [12]. The size of the dataset is determined by 54675 gene probes against 58 tumor tissue samples, which are divided into 18 tissue samples for SCC and 40 ones for AC. Once described the case study, we are going to show the results reached in each stage after applying the approach given in Figure 1, that is:

1. *Stage FP-I:* This stage also includes a data normalization process before applying the Mann-Whitney test. Then, after applying the test, ranking the dataset in ascending order for the p-value of each gene probe and selecting those gene probes whose p-value < 0.05, we have achieved an output dataset with 13141 probes to be passed to the next stage.
2. *Stage FP-II:* This stage takes as input, a dataset ordered by the p-value assigned to the probes. After that, it computes the variance and the value of function Score for each probe. Next, the current dataset has been ranked in descending order of the values given by Score. At this point, a threshold to make the filtering cutoff based on function Score has been fixed at 2.45, which is equivalent to 7.61% of probes in the current dataset. Finally, a dataset with 1000 gene probes has been given as output. Note that those probes present the greatest expression level variation against 58 tumor tissue samples at the same time that their statistical significance (the p-value) is high, too. The selected threshold has been chosen base on experience, for Score values higher than 2.45, no substantial changes noted on the achieved results.
3. *Stage CA:* This stage involves three processes consisting of setting the clustering methods to use, running and comparing the results for those methods in order to select the best one according to the used validity measures and finally, choosing a suitable clustering from the dendrogram of the selected method. Then, each of these processes has been performed in the following way:
   - *Settings:* The Euclidean distance between data has been used for all methods. In the case of HCGA, we must choose a fitness function and prefix values of a parameter set. So such a fitness function is based on the tradeoff of cluster homogeneity and separation to be defined as:

$$f_d(\mathfrak{G}) = \frac{1}{|\mathfrak{G}| - 1} \sum_{i=1}^{|\mathfrak{G}|-1} f_c(\mathfrak{C}_i), \tag{2}$$

   where $\mathfrak{G}$ is a dendrogram, $\mathfrak{C}_i$ is the clustering of level i in $\mathfrak{G}$ and f_c is the recurrent fitness function to evaluate a clustering of $\mathfrak{G}$, which is defined as,

$$f_c(\mathfrak{C}_{i+1}) = \frac{S_1^*(\mathfrak{C}_{i+1})}{g - k + 1} - \frac{\mathcal{H}_1^*(\mathfrak{C}_{i+1})}{k - 1} + \max \mathfrak{D}, \tag{3}$$

   where $S_1^*(\mathfrak{C}_{i+1})$ and $\mathcal{H}_1^*(\mathfrak{C}_{i+1})$ are separation and homogeneity for clustering $\mathfrak{C}_{i+1}$ respectively, being defined in [11], $k = |\mathfrak{C}_i|$ and $g = \binom{k}{2}$, being the

number of distances among the clusters of $\mathfrak{C}_{i+1}$. $\max \mathfrak{D}$ is the maximum distance from proximity matrix $\mathfrak{D}$ of the current dataset. Once introduced the fitness function, the HCGA parameters have been initialized as: crossover and mutation probability to 0.55 and 0.10 respectively, 30 individuals in the initial population and number of generations in $[10^3, 10^6]$. The crossover and mutation operators are given by default from the method [11].

– *Method comparison:* To compare the results of the three methods, we have used the cluster validity measures, homogeneity (Homog), separation (Separ) and silhouette width (SilhoW) [4], which have been applied to the dendrograms of each method . Keep in mind that the smaller the homogeneity value the higher the cluster quality, whereas the bigger the separation and silhouette width value the higher the cluster quality. Table 1 lists the scores reached by each method with respect to the validity measures, each score is the result of computing the mean cluster validity from the clusterings of each dendrogram. The standard error has also been shown for each score as well as the best scores for each index have been stressed. Then, from this table, we can conclude that the best result has been achieved by HCGA, which performs better than the others on two indices, i.e., separation and silhouette width.

Table 1 Comparison of cluster global validity based on separation and homogeneity for methods Eisen, HybridHclust and HCGA applied to the *lung cancer* dataset.

Method	Homog	Separ	SilhoW
Eisen	8.728 ± 0.108	13.018 ± 0.213	-0.026 ± 0.011
HybridHclust	$\underline{6.240 \pm 0.074}$	10.490 ± 0.038	0.077 ± 0.005
HCGA	6.435 ± 0.120	$\underline{10.657 \pm 0.070}$	$\underline{0.085 \pm 0.003}$

– *Clustering selection:* This stage selects a clustering from the HCGA dendrogram. This has firstly been made by selecting a clustering with the agglomerative coefficient $ac(\mathfrak{G}) = \arg_{i \in [1,|\mathfrak{G}|]} \max f_c(\mathfrak{C}_i)$ implemented in HCGA to select the best clustering according to homogeneity and separation. The above resulted in a clustering of 6 clusters, but it has been validated with respect to cluster visualizations given by tool 3D-VisualCluster [4], in which a clustering of 8 clusters has finally been selected. Such a clustering has been shown in Figures 2 and 3, through a visualization of dendrogram on heatmap and a 3D-scatterplot respectively, where point-genes in the same cluster have the same color. Note that these visualizations show cluster structures with high quality.

4. *Stage FP-III:* On the clustering selected in the stage above, this stage applies algorithm ClusterBoundery [4] to compute the boundary genes of each one of its clusters. Thus, this is the last step of the model stated by our proposal. Once completed the computation of boundary genes, we have achieved a set of 76 gene probes belonging to 63 genes, which have been assumed as

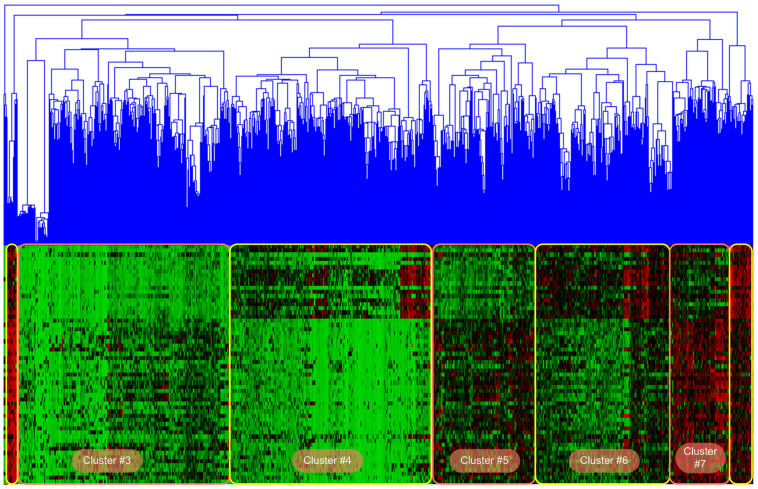


Fig. 2 Visualization of dendrogram with heatmap representing the clustering (8 clusters) selected from the HCGA dendrogram, for the *lung cancer* dataset.

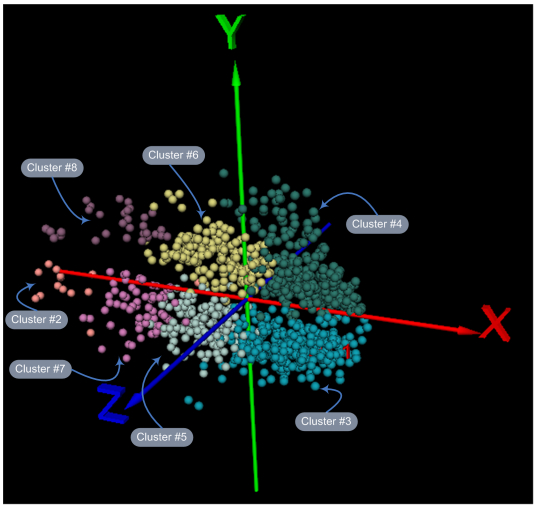


Fig. 3 Visualization of a 3D-scatterplot representing the clustering (8 clusters) selected from the HCGA dendrogram, for the *lung cancer* dataset.

the set of informative genes. According to the goal of this paper, we have that such subset of 63 informative genes can be used as the features of the 58 tumor tissue samples of the dataset. Consequently, a classifier can be trained only based on those genes. On the other hand, the following genes: {Nos1, MUC5AC, NAPSA, CEACAM6, IGF2BP3, S100A2, OIP5, UGT1A3, UGT1A3, ITGB6, SFTA3, CLDN3, RAB3B, PI3, MACC1,

LMO3, RASD1, XIST} among others from the found gene subset have previously been identified in other researches as genes related to lung cancer. This proves the biological validity of the proposed method when applied to microarray data.

4 Conclusions

The goal of this paper has been to provide a clustering-based method for gene selection from DNA-microarray data. Within this approach, the practical goal has been to target the selected genes to classification tasks in lung cancer. According to that, we have obtained a subset of 63 informative genes of which more than about 30% have been identified as related to lung cancer. Moreover, some of them have previously been identified as biomarkers of a lung cancer subtype. Hence, these promising results prove the reliability of our approach.

Acknowledgments This research has been co-financed by the European Social Fund (Operational Programme 2014-2020 for Castilla y León, EDU/128/2015 BOCYL) with regard to author Alfonso González-Briones.

References

1. Rothschild, S.I.: Advanced and metastatic lung cancer - what is new in the diagnosis and therapy. PRAXIS **104**, 745–750 (2015)
2. Wang, K.J., Melani, A., Chen, K.H., Wang, K.M.: A hybrid classifier combining borderline-SMOTE with AIRS algorithm for estimating brain metastasis from lung cancer: A case study in taiwan. Computer Methods and Programs in Biomedicine **119**, 63–76 (2015)
3. Berrar, D.P., Dubitzky, W., Granzow, M.: A Practical Approach to Microarray Data Analysis. Kluwer Academic Publishers, New York (2003)
4. Castellanos-Garzón, J.A., García, C.A., Novais, P., Díaz, F.: A visual analytics framework for cluster analysis of DNA microarray data. Expert Systems with Applications, Elsevier **40**, 758–774 (2013)
5. Lazar, C., Taminau, J., Meganck, S., Steenhoff, D., Coletta, A., Molter, C., deSchaetzen, V., Duque, R., Bersini, H., Nowé, A.: A survey on filter techniques for feature selection in gene expression microarray analysis. IEEE/ACM Transactions On Computational Biology And Bioinformatics **9**(4), 1106–1118 (2012)
6. Xia, C., Hsu, W., Lee, M.L., Ooi, B.C.: Border: Efficient computation of boundary points. IEEE Transactions on Knowledge and Data Engineering **18**, 289–303 (2006)
7. Jain, A.K., Murty, N.M., Flynn, P.J.: Data clustering: A review. ACM Computing Surveys **31**(3), 264–323 (1999)
8. Weiss, P.: Applications of generating functions in nonparametric tests. The Mathematica Journal **9**(4), 803–823 (2005)
9. Eisen, M., Spellman, T., Brown, P., Botstein, D.: Cluster analysis and display of genome-wide expression patterns. Proceedings of the National Academy of Sciences, USA **95**, 14863–14868 (1998)

10. Chipman, H., Tibshirani, R.: Hybrid hierarchical clustering with applications to microarray data. Biostatistics **7**, 302–317 (2006)
11. Castellanos-Garzón, J.A., Díaz, F.: An evolutionary computational model applied to cluster analysis of DNA microarray data. Expert Systems with Applications, Elsevier **40**, 2575–2591 (2013)
12. Kuner, R., Muley, T., Meister, M., Ruschhaupt, M., Buness, A., Xu, E., Schnabel, P., Warth, A., Poustka, A., Sltmann, H., Hoffmann, H.: Global gene expression analysis reveals specific patterns of cell junctions in non-small cell lung cancer subtypes. Lung Cancer **63**(1), 32–38 (2009)

Large-Scale Transcriptomic Approaches for Characterization of Post-Transcriptional Control of Gene Expression

Laura Do Souto, Alfonso González-Briones, Andreia J. Amaral, Margarida Gama-Carvalho and Juan F. De Paz

Abstract MicroRNAs are critical regulators of gene expression programs. It has been demonstrated that during the maturation of miRNAs some changes can happen leading to the production of isoforms called isomiRs. During our study, a PERL pipeline was developed to find all the miRs (miRNAs) and isomiRs in two sets of Next Generation Sequencing (NGS) of small RNA libraries derived from naïve and activated CD4+ T cells. Then, a differential expression analysis was performed using Bioconductor package DESeq. Our pipeline allowed us to find all the different types of isomiRs in both of conditions. Also, we found that the isoforms coming from changes on 3' are more frequent that in 5' ends. Tailing isoforms are described as the less frequent isomiRs. The use of DESeq on the read count dataset of these miRs and isomiRs identified a total of 5 miRs and 22 isomiRs which were differentially expressed. So, in addition to creating a new tool for isomiR analysis, we have been able to obtain evidence that upon activation miRs and isomiRs in T-cells are differentially expressed.

Keywords isomirs · Differentially expressed · NGS

L.D. Souto
UFR Sciences et des Techniques, University of Rouen, Mont Saint Aignan Cedex, France
e-mail: dosouto.laura@univ-rouen.fr

A. González-Briones(✉) · J.F. De Paz
Biomedical Research Institute of Salamanca/BISITE Research Group,
University of Salamanca, Edificio I+D+i, 37008, Salamanca, Spain
e-mail: {alfonsogb,fcofds}@usal.es

A.J. Amaral · M. Gama-Carvalho
BioFIG-Centre for Biodiversity, Functional and Integrative Genomics, Faculty of Science,
University of Lisbon, Lisbon, Portugal
e-mail: {adfonseca,mhcarvalho}@fc.ul.pt

M.S. Mohamad et al. (eds.), *10th International Conference on PACBB*,
Advances in Intelligent Systems and Computing 477,
DOI: 10.1007/978-3-319-40126-3_12

1 Introduction

The discovery of microRNAs is quite recent. We only are starting to understand the mechanisms that contribute to the production of miRs and their biological functions are still debated. Some studies have shown that during microRNA maturation some changes can occur leading to the production of miR isoforms called isomiRs. Three main types of changes can occur: 1) the 5' or 3' end position of the mature miR can be altered, leading to the gain or loss of (usually) one or two nucleotides; 2) tailing of the miRs can occur by addition of nucleotides; and 3) the internal sequence of the miR may vary. These changes can affect the seed or downstream sequence and change the repertoire of miRNA targets, consequently having impact on the gene expression program of a cell [10, 7].

Recent studies have shown that the first type of isoform (5' and 3' end isoforms) can be produced by a shift in a cleavage sites by Drosha or Dicer or through the action of 5' or 3' end trimming enzymes (Fig. 1) [20, 5, 15]. Additionally, miR isoforms may be generated by differential processing of paralogous miR genes. The second type of isoform are produced through the addition of one to several nucleotides, generating homopolymeric tails, by Terminal Nucleotidyl Transferases [21]. Studies have further shown that the number of adding nucleotides does not exceed 5 [8, 20]. These nucleotides are usually As or Us and although the impact of their addition is not well understood, it is known to occur in some cases in association with miR turnover processes. Finally internal substitutions can be caused by RNA editing enzymes, the best characterized of which is ADAR, a double-stranded RNA-specific adenosine deaminase which is able to edit A into I, see Fig. 1, which is converted to G during reverse transcription.

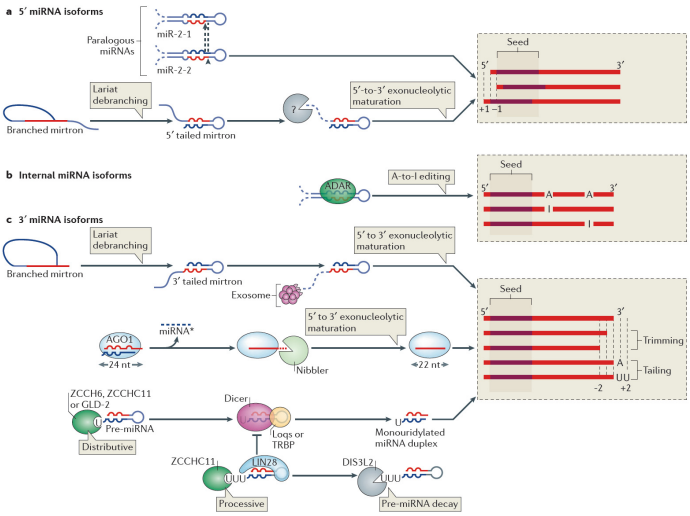


Fig. 1 Mechanisms originating microRNA isoforms

The availability of small-RNA-seq enables the description and investigation of all isomiRs that are being expressed in a tissue, contributing to the understanding of their biological relevance. However, the most widely used tools for the analysis of next generation sequencing (NGS) data do not allow the analysis of data at this level of complexity, and the few published tools specific for isomiR analysis do not allow the accurate detection of all possible forms [14]. In this project we aim to develop a user friendly pipeline that allow the identification of all reported types of isomiRs in NGS data. Furthermore, we aim to apply this tool to the analysis of a specific NGS dataset in order to unravel the impact of the TCR-dependent activation of CD$+T cells (HIV target cells) in miR biogenesis, considering preliminary results that indicate a potencial increase of isomiRs in response to TCR stimulation.

2 Material and Methods

Experiment Design

A study was conducted in which blood samples of 9 healthy donors were obtained and, naive CD4+ T cells were purified and stimulated with antibodies against the T cell receptor (TCR), to mimic the activation by an antigen.

Total RNA was extracted from unstimulated and stimulated CD4+ T cells preserving the small RNA fraction. Molecules ranging between 15 and 30 nucleotides were purified and used to generate two small RNA-seq libraries that were sequenced on an Illumina Genome analyser II, yielding twenty million and sixteen millions of reads, provided in fastq format. This structure was changed into a table format with seq id, read, and quality score data and quality filters were applied in order to discard reads with an average score lower than 20, with homopolymers longer than half the read size or with unknown bases using in house developed PERL scripts. This processed eliminated 1% of reads.

Preparing the Database

MiRbase is a public repository of all known microRNAs and their annotations [17]. miRbase gives researchers without a bioinformatics background a user friendly access, enabling users to perform different types of queries by miR accession number, name or keyword, by genomic location, by tissue expression or by sequence, but also by cluster of miR with a chosen distance. miRbase further allows the download of the sequences of all the repository of pre-miRs, mature miRs and mature star miRs in fasta format.

However, in order to investigate the existence of all the different possible types of isomiRs, information regarding the miR coordenades within a pre-miR is required and for an efficient query this information should be organized within the same input file.

The fasta file of all human pre-miRs and miRs available in miRbase version 20 was downloaded and a PERL script was developed that retrieves the miR coordenates for each pre-miR. Since some pre-miRs generate two functional miRs, the 5p and 3p miRs, two database files were created, hairpin DB 3 and hairpin DB 5. These files contain the identifier (id) of the hairpin, its sequence, the id of the mature miR, sequence, length, start on the hairpin and its end position.

Identifying isomiRs with Additions and / or Trimmings

This script was reading these two files: one file holding the reads and its frequency and a hairpin DB file and was searching for total homology of a read within a pre-miR. If there was a match, the script first interrogates if it matches ambiguously or not. Then it will provide us the coordinates of the read on the hairpin, classify a read as mature if it overlapped the coordenades of a mature miR and if it does, the output provides information regarding the number of nucleotides trimmed or added in comparison with the mature miR coordinates (Fig. 2)

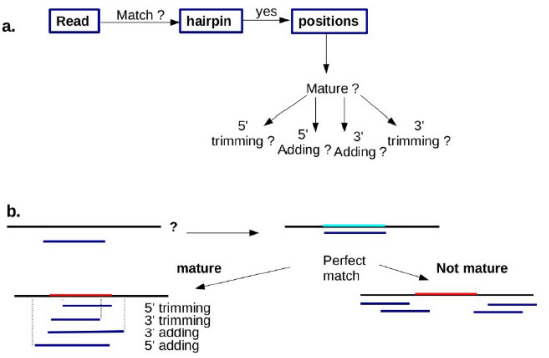


Fig. 2 Find IsomiRs workflow

Identifying Differentially Expressed Isoforms

After identifying isomiRs, the following question is if these are differentially expressed between two or more experimental conditions. In this study we aim to investigate if activation of CD4+ T cells has an impact on the biogenesis of miRNAs. So the first step to perform this analysis consists in generating a table of counts. A PERL script was developed, which outputs a table with the following structure: hsa-miR-146b-5p:0 0 0 0 0 0 1 C/T 21 623 1095.

First, we have the name of the miRNA before the ":", then the indication of no trimming, no shift, no tailing in 3' and the same in 5', after this the identification of 1 change a C to T at position 21. Then, on the next columns we have the counts within each experimental condition; this isomiRs is found 623 times in naive cells and 1095 times in activated ones.

Also, summary statistics regarding isomiR diversity were queried from the count table using bash commands.

Then, read count normalization and testing for differential expression was performed using DESeq package of Bioconductor within R statistical environment using the previous table [6, 9, 3].

3 Results

Our analysis allowed to identify as miRNAs 70% of reads, from these, more than 97% were uniquely assigned to a canonical miRNA and 3% to another region of the pre-miR. Reads that do not overlap the region of pre-miRs from which mature miRs derive most likely degradation products of these transcripts, and were therefore discarded from downstream analysis.

Type of Isomirs

We next investigated the frequency of each type of isomiRs. Results displayed in Fig. 3 show that isomiRs of types 3' trimming and 3' additions were most frequent both in naive and activated CD4+ T cells. IsomiRs with homopolymeric tailings displayed the lowest frequency along with IsomiRs holding internal mismatches. Finally an important observation is that modifications of 3'ends occurred at a higher frequency than 5' modifications.

Considering differences between activated and naive CD4+ T cells that activated CD4+ T cells displayed a lower occurrence of 4% of isomiRs with 3' extensions (Fig. 3).

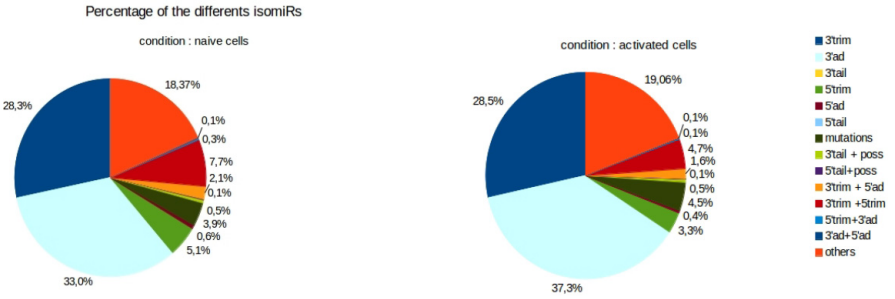


Fig. 3 Difference of miRNA isoforms type

Since most reads displayed a length between 20 and 24 nucleotides we further focused our study on these in order to understand in better detail the changes between the mature and isomiR sequence. As may be observed in Fig. 4, reads with a length of 22 nucleotides are the only ones displaying a higher frequency of canonical miRs, whereas in the remaing read lengths, isomiRs were the most frequent.

Focus on read lengths 21-24.

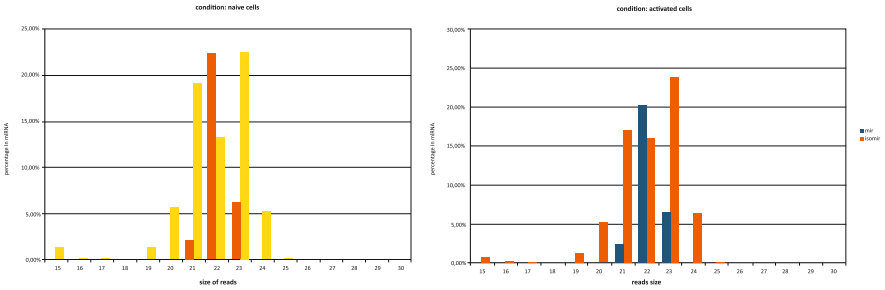


Fig. 4 Percentage of miRs and isomiRs by read size

The most frequent length of modifications in the 3' end extensions depend on read length of the isomiR being mostly of one nucleotide and of two in the case of 24 nucleotides reads. For 3' trimming the most frequently observed is change of one nucleotide. Regarding the 5' end for read of lengths 21-24 we mainly observed extensions or trimmings of one nucleotide (Fig. 5).

We further observed that in 21 nucleotide reads, the most frequent isomiRs are 3' trimming isomiRs and 3' plus 5' trimming isomiRs. IsomiRs with 3' trimming were the most frequent, and more abundant in activated CD4+ T cells, while trimming on both ends was more frequent in naive than activated CD4+ T cells.

For the 22 nucleotide reads, 3' extension isomiRs were the most frequent. All isomiRs were more frequent in activated than naive CD4+ T cells except for 5' trimming isomiRs, which were most frequent on naive cells.

Finally, for the 23 and 24 nucleotide reads we observed a majority of 3' extensions isomiRs and a few isomiRs with mismatches in the 23 nucleotide reads. Only for reads with 24 nucleotides we observed change between the two conditions: the isomiRs are more frequent in activated than in naive cells.

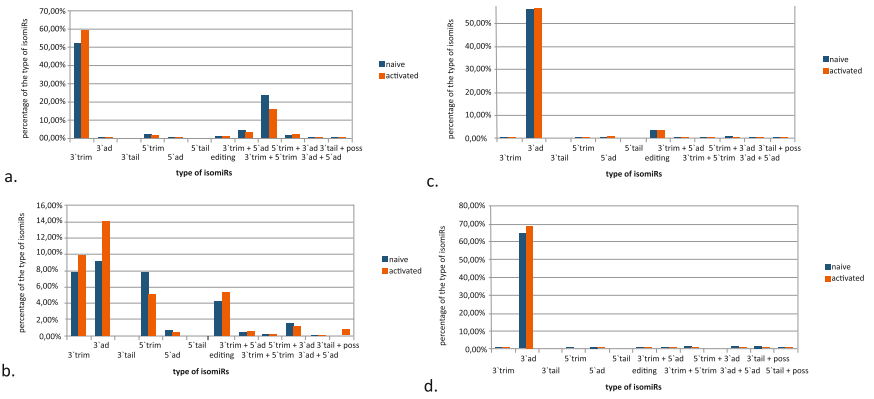


Fig. 5 Percentage of miRs type

Also, for the trimming and adding we found that for most of the data one base is trimmed or added. Only for the 24 nucleotides reads we found more adding of two bases than one.

Differential Expression Analysis with DESeq

In Fig. 6 the correlation between log2 fold change of each isomiR and the mean of the corresponding normalized counts between naive and activated T cells is shown. We can observe a high dispersion in highly expressed and in low expressed miRs and isomiRs.

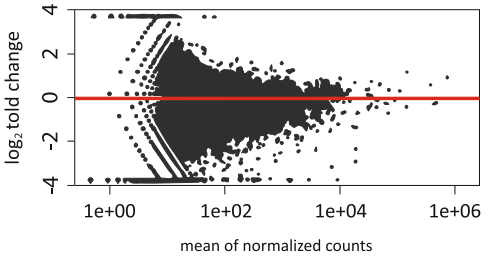


Fig. 6 MA plot for read with 21 to 24 nucleotides length

ID	baseMean	baseMeanA	baseMeanE	Change	FoldChange	pval	padj
hsa-let-7a-5p:0_0_0_0_0_0_0	**216024,79**	**392452,15**	**39597,43**	**0,1**	**-3,31**	**0,02**	**1**
hsa-let-7a-5p:0_0_0_0_1_0_0	341,56	643,57	39,55	0,06	-4,02	0	0,63
hsa-let-7a-5p:0_0_0_1_0_0_0	247,55	478	17,1	0,04	-4,8	0	0,21
hsa-let-7a-5p:0_1_0_0_0_0_0	9033,23	16787,88	1278,58	0,08	-3,71	0	0,99
hsa-let-7a-5p:0_5_0_5_0_0_0	44,16	85,12	3,21	0,04	-4,73	0	0,81
hsa-let-7a-5p:1_0_0_0_1_0_0	399,82	770,78	28,86	0,04	-4,74	0	0,21
hsa-let-7a-5p:2_0_0_0_1_0_0	42,63	84,19	1,07	0,01	-6,3	0	0,53
hsa-let-7b-5p:0_0_0_0_0_0_0	**18709,16**	**32966,81**	**4451,5**	**0,14**	**-2,89**	**0,01**	**1**
hsa-let-7b-5p:0_0_0_0_0_0_1_T/C_22	199,58	371,36	27,8	0,07	-3,74	0	0,85
hsa-let-7b-5p:0_0_0_0_0_0_1_T/C_22	30	58,93	1,07	0,02	-5,78	0	0,81
hsa-let-7c-5p:0_0_0_0_0_0_0	2898,78	5387,05	410,51	0,08	-3,71	0	0,81
hsa-let-7c-5p:0_1_0_0_0_0_0	182,94	342,36	23,52	0,07	-3,86	0	0,81
hsa-let-7c-5p:0_1_0_0_0_0_2_T/A_31_T	45,57	87,93	3,21	0,04	-4,78	0	0,81
hsa-let-7d-5p:0_0_0_0_0_0_0	**12886,54**	**23329,23**	**2443,84**	**0,1**	**-3,25**	**0**	**1**
hsa-let-7d-5p:0_1_0_0_0_0_0	1298,56	2444,24	152,87	0,06	-4	0	0,63
hsa-let-7f-5p:0_0_0_0_0_0_0	**12886,54**	**23329,23**	**2443,84**	**0,1**	**-3,25**	**0**	**1**
hsa-let-7f-5p:0_0_0_0_0_0_1_G/T_11	109,91	204,86	14,97	0,07	-3,77	0	0,99
hsa-let-7f-5p:0_0_0_1_0_0_0	301,67	573,41	29,93	0,05	-4,26	0	0,53
hsa-let-7g-5p:0_0_0_0_0_0_0	**156492,01**	**223169,28**	**89814,74**	**0,4**	**-1,31**	**0,31**	**1**
hsa-let-7g-5p:1_0_0_0_1_0_0	44,16	85,12	3,21	0,04	-4,73	0	0,81
hsa-miR-1:0_0_0_0_0_0_0	**16752,67**	**30948,18**	**2557,16**	**0,08**	**-3,6**	**0**	**1**
hsa-miR-1:0_0_0_0_0_0_1_T/A_67	32,34	63,61	1,07	0,02	-5,89	0	0,81
hsa-miR-1:0_0_0_0_0_0_1_T/A_73	93,14	177,73	8,55	0,05	-4,39	0	0,81
hsa-miR-1:0_1_0_0_0_0_0	310,76	588,38	33,14	0,06	-4,15	0	0,61
hsa-miR-10a-5p:0_0_0_0_0_0_0	**1707,8**	**2679,03**	**736,57**	**0,27**	**-1,86**	**0,05**	**1**
hsa-miR-10a-5p:1_0_0_0_1_0_0	19,18	38,35	0	0	Inf 1	0	0,85
hsa-miR-29b-1-5p:0_0_0_0_0_0_0	386,13	708,11	64,14	0,09	-3,46	0	0,99
hsa-miR-369-3p:0_0_0_0_0_0_0	1403,59	2612,61	194,57	0,07	-3,75	0	0,81
hsa-miR-382-5p:0_0_0_0_0_0_0	3255,44	6006,3	504,59	0,08	-3,57	0	0,99
hsa-miR-382-5p:1_0_0_0_0_0_0	38,02	73,9	2,14	0,03	-5,11	0	0,81
hsa-miR-409-3p:0_0_0_0_0_0_0	**770,78**	**1354,48**	**187,08**	**0,14**	**-2,86**	**0**	**1**
hsa-miR-409-3p:0_1_0_0_0_0_0	180,33	333,94	26,73	0,08	-3,64	0	0,99
hsa-miR-7641:0_5_0_0_0_0_1_T/C_2	66,15	1,87	130,42	69,71	6,12	0	0,21
hsa-miR-98-5p:0_0_0_0_0_0_0	61,2	115,99	6,41	0,06	-4,18	0	0,99
hsa-miR-98-5p:1_0_0_0_0_0_0	619,31	1147,75	90,87	0,08	-3,66	0	0,81

Fig. 7 MiRNAs and isomiRs differentially expressed (not in bold)

After testing for differential expression, we obtained a table of five miRs and twenty-two isomiRs with p-value<0.05 although significance was lost after correcting for multiple testing. Nevertheless, we can observe that many log2 fold changes between 4 to 6 (see Fig. 7). Interestingly we observed in some cases that the expression of isomiRs changed according with activation of CD4+ T cells, whereas the canonical form does not.

4 Discussions

The aim was the development of a pipeline to identify isomiR molecules in NGS data that is user friendly for users with basic bioinformatics skills. Although a tool is currently available [14] it does not allow to identify all reported types of isomiRs [22]. A PERL script was developed that allows accurate detection of 5' and 3' adding, 5' and 3' tailing, 5' and 3' trimming, and internal editings. The developed script required high allocation of memory. Although it allows to accurately detect all possible types of isomRs.

The percentage of reads annotated as deriving from pre-mir loci (70%) was similar to several previous studies using small-RNA-seq [18, 11, 4, 12]. Regarding the proportion of the different types of isomiRs found among our data, we found that isoforms with extensions in the 3' end were more frequent than the ones in the 5' end. This is accordance with previous studies [16]. We found that the predominant type of isomiRs are 3' extensions and 3' trimming in both of conditions as it was observed the presence of these isomiRs was more frequent in activated than in naive T cells ([23]), a phenomena which has been observed in epidermial cells under differentiation [19].

Regarding isomiRs holding mismatches, the overall observed frequency was in magnitude similar to the occurrence of sequencing errors and in accordance with a recent study [1]. Moreover, similar to this study the most frequent type of substitutions was G to A and T to A. This is somehow contradicting with previous studies and it does not correspond to the type of editing events related with the activity of ADAR protein which would produce a type of substitutions of A to I which would be interpreted by the base caller algorithm as a G [13]. Another interesting point is the fact that these events occur mostly in the 3' end of reads, regions which are known to be higher prone to sequencing errors. Furthermore the most frequent location is the last nucleotide, which besides holding a lower quality score can also deviate from a nucleotide belonging to the adapters that was not trimmed. This possibilities need to be discarded before we can conclude from these reads which are effectively derived from RNA editing events. In particular considering that these potential editing events were observed at the same frequency of sequencing errors.

In our study, the majority of isomiRs and miRs potentially differentially expressed were down regulated in activated cells. In fact, after going on OMIM database, we found that these miRNAs are implicated in the regulation of targets involved with cell growth (hsa-let-7, hsa-miR-369, hsa-miR-382), proliferation

(hsa-miR-1) and control proteins involved in immune escape (hsa-miR-29b, hsa-miR-98). Only one isomiR derived from hsa-miR-764 was potentially up-regulated in activated cells, and is involved in the post-transcriptional control of CXCL1 protein which is involved in angiogenesis but also in the growth of malignant cells (OMIM database).

So, in spite of the fact that the analysis of differential expression could not retrieve any significant result after correction for multiple testing, nevertheless we could identify isomiRs which are strongly down-regulated upon activation o T cells (-4 to -6 fold changes) and which are involved in relevant biological processes within this context. We believe that the lack of significance after multiple testing is due to the lack of biological replicates combined with an over-correction of multiple testing because the algorithm encoded within DESeq does not allow to test for which isomiRs within a miRNA are enriched. A similar algorithm, DEXSeq allows to perform this testing, however it requires the existence of biological replicates.

5 Conclusions

This study has shown that it is possible to generate a pipeline for the identification of all types of isomiRs that using a lower capacity processing server is able to uncover another level of complexity of miRNAs.

Also, although the pipeline allows the detection of all types of isomiRs, the accurate identification of isomiRs holding internal editings is still not effective. Although we can identify reads that may correspond to this type of events, our analysis does not allow to separate between other events that could originate the same outcome, sequencing errors, inefficient adaptor trimming, single nucleotide polymorphisms. Including a variant call algorithm that based on a mapping of these reads could infer a probability to this type of events is envisaged in the near future.

Overall, the work performed has contributed to the generation of a novel tool for the study of miR biology that was so far lacking and which is predicted to have positive impact for future studies in the field.

Acknowledgements The research of Alfonso González-Briones has been co-financed by the European Social Fund (Operational Programme 2014-2020 for Castilla y León, EDU/128/2015 BOCYL).

References

1. Ameres, S.L., Zamore, P.D.: Diversifying microRNA sequence and function. Nature Reviews Molecular Cell Biology **14**(8), 475–488 (2013)
2. Ameres, S.L., Horwich, M.D., Hung, J.H., Xu, J., Ghildiyal, M., Weng, Z., Zamore, P.D.: Target RNA–directed trimming and tailing of small silencing RNAs. Science **328**(5985), 1534–1539 (2010)

3. Anders, S., Huber, W.: Differential expression analysis for sequence count data. Genome Biol. **11**(10), R106 (2010)
4. Babiarz, J.E., Ruby, J.G., Wang, Y., Bartel, D.P., Blelloch, R.: Mouse ES cells express endogenous shRNAs, siRNAs, and other Microprocessor-independent, Dicer-dependent small RNAs. Genes & Development **22**(20), 2773–2785 (2008)
5. Berezikov, E., Robine, N., Samsonova, A., Westholm, J.O., Naqvi, A., Hung, J.H., Okamura, K., Dai, Q., Bortolamiol-Becet, D., Martin, R., Zhao, Y.: Deep annotation of Drosophila melanogaster microRNAs yields insights into their processing, modification, and emergence. Genome Research **21**(2), 203–215 (2011)
6. Bickel, D.R.: Degrees of differential gene expression: detecting biologically significant expression differences and estimating their magnitudes. Bioinformatics **20**(5), 682–688 (2004)
7. Chiang, H.R., Schoenfeld, L.W., Ruby, J.G., Auyeung, V.C., Spies, N., Baek, D., Johnston, W.K., Russ, C., Luo, S., Babiarz, J.E., Blelloch, R.: Mammalian microRNAs: experimental evaluation of novel and previously annotated genes. Genes & Development **24**(10), 992–1009 (2010)
8. Dreyfus, M., Régnier, P.: The poly (A) tail of mRNAs: bodyguard in eukaryotes, scavenger in bacteria. Cell **111**(5), 611–613 (2002)
9. Dudoit, S., Gentleman, R.C., Quackenbush, J.: Open source software for the analysis of microarray data. Biotechniques **34**(13), S45–S51 (2003)
10. Ebhardt, H.A., Tsang, H.H., Dai, D.C., Liu, Y., Bostan, B., Fahlman, R.P.: Meta-analysis of small RNA-sequencing errors reveals ubiquitous post-transcriptional RNA modifications. Nucleic Acids Research **37**(8), 2461–2470 (2009)
11. Eipper-Mains, J.E., Eipper, B.A., Mains, R.E.: Global approaches to the role of miRNAs in drug-induced changes in gene expression. non-coding RNA and Addiction **33** (2012)
12. Esteller, M.: Non-coding RNAs in human disease. Nature Reviews Genetics **12**(12), 861–874 (2011)
13. George, C.X., Gan, Z., Liu, Y., Samuel, C.E.: Adenosine deaminases acting on RNA, RNA editing, and interferon action. Journal of Interferon & Cytokine Research **31**(1), 99–117 (2011)
14. Giurato, G., De Filippo, M.R., Rinaldi, A., Hashim, A., Nassa, G., Ravo, M., Rizzo, F., Tarallo, R., Weisz, A.: iMir: an integrated pipeline for high-throughput analysis of small non-coding RNA data obtained by smallRNA-Seq. BMC Bioinformatics **14**(1), 362 (2013)
15. Glazov, E.A., Cottee, P.A., Barris, W.C., Moore, R.J., Dalrymple, B.P., Tizard, M.L.: A microRNA catalog of the developing chicken embryo identified by a deep sequencing approach. Genome Research **18**(6), 957–964 (2008)
16. Guo, L., Zhang, H., Zhao, Y., Yang, S., Chen, F.: Selected isomiR expression profiles via arm switching? Gene **533**(1), 149–155 (2014)
17. Kozomara, A., Griffiths-Jones, S.: miRBase: integrating microRNA annotation and deep-sequencing data. Nucleic Acids Research, gkq1027 (2010)
18. Landgraf, P., Rusu, M., Sheridan, R., Sewer, A., Iovino, N., Aravin, A., Pfeffer, S., Rice, A., Kamphorst, A.O., Landthaler, M., Lin, C.: A mammalian microRNA expression atlas based on small RNA library sequencing. Cell **129**(7), 1401–1414 (2007)
19. Llorens, F., Hummel, M., Pantano, L., Pastor, X., Vivancos, A., Castillo, E., Mattlin, H., Ferrer, A., Ingham, M., Noguera, M., Kofler, R.: Microarray and deep sequencing cross-platform analysis of the mirRNome and isomiR variation in response to epidermal growth factor. BMC Genomics **14**(1), 1 (2013)

20. Rissland, O.S., Mikulasova, A., Norbury, C.J.: Efficient RNA polyuridylation by noncanonical poly (A) polymerases. Molecular and Cellular Biology **27**(10), 3612–3624 (2007)
21. Song, M.G., Kiledjian, M.: 3' Terminal oligo U-tract-mediated stimulation of decapping. Rna **13**(12), 2356–2365 (2007)
22. Stokowy, T., Eszlinger, M., Świerniak, M., Fujarewicz, K., Jarząb, B., Paschke, R., Krohn, K.: Analysis options for high-throughput sequencing in miRNA expression profiling. BMC Research Notes **7**(1), 144 (2014)
23. Vickers, K.C., Sethupathy, P., Baran-Gale, J., Remaley, A.T.: Complexity of microRNA function and the role of isomiRs in lipid homeostasis. Journal of Lipid Research **54**(5), 1182–1191 (2013)

Part III
Genomics

FLAK: Ultra-Fast Fuzzy Whole Genome Alignment

John Healy

Abstract The advent of high throughput DNA sequencing has lead to the availability of a rapidly growing number of genomes of complete or draft quality. Whole genome alignment has consequently become an increasingly important field in bioinformatics. This paper describes a novel approach for comparing two whole genomes based on fuzzy logic. Benchmarks against pre-eminent whole genome alignment systems have demonstrated that the fuzzy approach outperforms existing systems in the context of alignment times and enables analyses that are not possible with other approaches.

Keywords Whole genome alignment · Fuzzy · Rapid · *k*-mer · hedges

1 Introduction

The manipulation of biological sequence data using sequence alignment techniques is fundamental to the solution of many problems in bioinformatics. A key requirement of any alignment system is the capability of performing an *in-exact* global or local alignment of sequences, as evolutionary events give rise to variability in related sequences that manifest themselves, at a micro level, as polymorphisms, insertions and deletions (indels). At a macro level, segmental duplications, deletions, inversions and transpositions may result in major structural variability, even within the same species. Whole Genome Alignment (WGA) methods have been reported by a number of authors [1, 2, 3, 4, 5] and have proven themselves indispensable for understanding how selective evolutionary forces may act across genomes [6].

J. Healy(✉)
Department of Computer Science & Applied Physics, School of Science & Computing, Galway-Mayo Institute of Technology, Galway, Ireland
e-mail: john.healy@gmit.ie

M.S. Mohamad et al. (eds.), *10th International Conference on PACBB*,
Advances in Intelligent Systems and Computing 477,
DOI: 10.1007/978-3-319-40126-3_13

With thousands of genomes now being sequenced each year and at a rapidly decreasing cost [7], there is a cogent need for a number of different bioinformatics tools designed for whole-genome comparison and analysis. This paper describes a novel approach to WGA and analysis based on fuzzy logic. FLAK (**F**uzzy **L**ogic **A**nalysis of **k**-mers) is a software system designed to perform a fast fuzzy whole-genome comparison of two DNA sequences and enable fuzzy operations to be performed on a finished alignment in a visual and intuitive way. FLAK is written in Java and uses a 2-bit encoding mechanism (A=00, C=01, G=10, T=11) to represent and compress each 32-mer substring of a genome into a single 64-bit (8 byte) primitive type. The representation of DNA sequence information as a bit vector greatly reduces the space complexity of the system and enables FLAK to scale from small prokaryotic genomes (<5Mbps) to large mammalian chromosomes or genomes (>3 Gbps). The software is freely available from http://www.flakbio.com.

2 Fuzzy *k*-mer Matching

Fuzzy logic is the theory of fuzzy sets, sets that describe vagueness. In the context of biological sequence alignment, the vagueness in question relates to the degree of homology between two sequences. As FLAK is designed around fuzzy sets and fuzzy logic, the alignment system is fundamentally different to all existing genome aligners and deals with degrees of fuzzy set membership and degrees of homology. All existing whole-genome aligners are based on bivalent boolean logic and, at an abstract level, ask the question *"Are the sequences similar?"*. In contrast, the fuzzy nature of FLAK is designed to answer the question *"how similar are the sequences?"*.

2.1 FLAK Alignment Process

FLAK uses a fuzzy hash map [8] to index a reference genome, enabling approximately matching k-mer sequences to be grouped together into fuzzy sets, while also enabling an average $\mathcal{O}(1)$ time complexity for insertion, deletion and search operations. The reference sequence is first decomposed into a tiling of 32-mers and loaded into a fuzzy hash map with a user-specified consecutive or spaced-seed. The space complexity of the fuzzy hash map is in the order of $\mathcal{O}(\frac{G}{32-i})$, where G is the size of the reference genome and i is the degree of overlap between adjacent 32-mers. The full spectrum of 32-mers, offset by one base, is then extracted from the query sequence. Each 32-mer is aligned in $\mathcal{O}(1)$ time against the reference sequence in the fuzzy hash map. Contiguous 32-mers are chained together, with chains having a length greater than a user-specified threshold reported as matches.

Table 1 Edit distances and their corresponding fuzzy membership values.

Edit Distance	$\mu_A(x)$	Edit Distance	$\mu_A(x)$	Edit Distance	$\mu_A(x)$
0	1	11	0.66	22	0.31
1	0.97	12	0.63	23	0.28
2	0.94	13	0.59	24	0.25
3	0.91	14	0.56	25	0.22
4	0.88	15	0.53	26	0.19
5	0.84	16	0.50	27	0.16
6	0.81	17	0.47	28	0.13
7	0.78	18	0.44	29	0.09
8	0.75	19	0.41	30	0.06
9	0.72	20	0.38	31	0.03
10	0.69	21	0.34	32	0

2.2 Crisp v/s Fuzzy Alignments

In classical set theory, membership of a **crisp** set A of X is defined as a function $f_A(x) : X \rightarrow 0, 1$, where $f_A(x) = 1$ if $x \in A$, called the **characteristic function** of A. Membership of a classical crisp set is therefore predicated on a boolean characteristic function, requiring an exact match to align two subsequences. However, a fuzzy set A of X can be defined by a function $\mu_A(x)$, called the **membership function** of A, which spans the continuum of real numbers in the interval [0..1]:

$$\mu_A(x) : X \rightarrow [0 \ldots 1] \text{ where } \begin{cases} \mu_A(x) = 1 & \text{if } x \text{ is totally} \in A \\ \mu_A(x) = 0 & \text{if } x \text{ is not} \in A \\ 0 < \mu_A(x) < 1 & \text{if } x \text{ is partly} \in A \end{cases}$$

Using this approach, approximately matching k-mer subsequences, 32-mers in the FLAK system, can be grouped together and accessed rapidly in a fuzzy hash map. The membership function $\mu_A(x)$ is implemented using an optimised variation of the Levenshtein distance algorithm [9]. For a query sequence S and a reference sequence T, both of size k (32-mers), the membership value returned by the algorithm is $\mu_A(x) = 1 - levenshtein(S, T)/k$. Although, in common with other approximate string-matching algorithms based on dynamic programming, the Levenshtein distance has a space and time complexity of $\mathcal{O}(n^2)$, for each 32-mer search the algorithm is only applied to the fuzzy sets in the matching bucket of a hash map. A table of the possible Levenshtein distances for a 32-mer and their corresponding fuzzy set membership values is shown below.

FLAK permits the parametrisation of an alignment with any type of seed with a seed weight ≥ 8 and a length ≤ 32 and provides a selection of consecutive and spaced-seeds based on those recommended by Choi *et al* [10] and by Mac and Benson [11]. In contrast with other genome aligners, which are based on exact-matching data structures like suffix trees or use seed-and-extend heuristics, FLAK uses fuzzy seeds (fuzzy k-mers) to accommodate an approximate k-mer match.

Approximate or fuzzy matching enables FLAK to detect alignments in the presence of polymorphisms and indels that are common in DNA sequences and reduce the sensitivity of exact-matching approaches.

2.3 Seed Representation

FLAK processes sequences in chunks of 32 bases (32-mers) and uses a fuzzy hash map to provide the speed required for an approximate 32-mer match of a large amount of sequence data. Consider the crisp spaced seed 111010010100110111, with a length of 18 and a seed weight of 11, used by PatternHunter [12]. The positions in the seed denoted by 1s are must-match indices, with 0s indicating don't-care indices. This seed is defined in FLAK as ###-#--#-#--##-###, with a hash character (#) representing a must-match index and dashes (-) denoting **may-care** positions. FLAK will execute a 32-mer alignment using this seed in a two-step operation:

1. **Hashing:** An integer value is computed from the hash positions of the must-match indices in the seed. E.g., the 32-mer, *S*, with the sequence TGTACCGGATATGACGTTACGGATAGGCCAAA will be processed as follows:

```
TGTACCGGATATGACGTTACGGATAGGCCAAA   Query 32-mer
--------------###-#--#-#--##-###   Fuzzy Seed
--------------CGT-A--G-T--GC-AAA   Masked 32-mer
```

   where $hash(S) = \sum_{i=0}^{n=1} char(S) \times 31^{n-(i+1)}$, i.e. an integer value computed from the underlying ASCII values of the masked 32-mer. The search bucket index of a fuzzy hash map of size n can be computed using the modulus method, $index = hash(S) \bmod n$.

2. **Approximate 32-mer Comparison:** If the search bucket in the fuzzy hash map has one or more entries, a dynamic programming algorithm will compute the Levenshtein edit distance between candidate matches. Any matches at or above a user specified fuzzy threshold, the β cut-off, will be processed. For example, comparing the query 32-mer TGTACCGGATATGACGTTACGGATAGGCCAAA against the reference 32-mer GGTACCGGATATGACGTAAAAGTTCCGCGAAA in the fuzzy hash map will produce a match for β=0.75 (an edit distance of 8):

```
TGTACCGGATATGACGTTACGGATAGGCCAAA   Query 32-mer
GGTACCGGATATGACGTAAAAGTTCCGCGAAA   Reference 32-mer
*GTACCGGATATGACGT*A**G*T**GC*AAA   Edit distance = 8
```

   The edit distance of 8 will yield a fuzzy value of 0.75, as $\mu_A(x) = 1 - levenshtein(S, T)/k = 1 - (8/32) = 0.75$.

2.4 Controlling Sensitivity and Specificity

The sensitivity of any hash-based alignment mechanism that uses crisp seeds is determined by the seed weight, i.e. the number of hash indices in the seed. For crisp consecutive and spaced seeds, the seed weight also controls the specificity of a k-mer match and the overall running time of an alignment. As the seed weight increases, so too does the specificity of a k-mer match. Increasing the seed weight from 9 to 13 will have a significant impact on the speed, sensitivity and specificity of a crisp k-mer alignment. For the fuzzy seeds used by FLAK, the seed weight controls both alignment speed and sensitivity, but does not determine specificity. The specificity of a fuzzy seed is determined by the β cut-off threshold, a fuzzy value between 0 and 1. For a 32-mer sequence, using the seed ###-#–#-#–##-### with a β cut-off of 0 will produce the exact same set of k-mer matches as a crisp spaced seed, i.e. the same sensitivity and specificity. Increasing the β cut-off to 1, will restrict matches to 32-mers with an edit distance of zero (an exact 32-mer match). Selecting a β cut-off of 0.3 will yield a specificity of 20%, while a value of 0.8 will have a specificity of approximately 92%. The exact relationship between β and specificity depends on the k-mer spectrum of a genome, but will form an "S" curve. Typical usage will have a β cut-off threshold of 0.75 or greater.

3 Fuzzy Operations

By utilising a fuzzy model to represent and compare DNA sequences, the full range of fuzzy set and logical operations can be exploited by FLAK. To date, no other biological sequence comparison software has been published that uses fuzzy logic in its operation. Mathematically, fuzzy logic is a superset of bivalent boolean logic and, as a consequence, the logical operators used in boolean logic also have the same truth values in fuzzy logic. Fuzzy set operations are known as *hedges* and are used to strengthen or weaken set membership. In FLAK, these operations are applied to an alignment *post hoc*, using simple GUI controls, and can be used to identify areas of strong or weak homology. The result of a fuzzy operation is visualised and affects the output of alignment data. The operations described below are applied to the result of the $\mu_A(x)$ membership function returned by the Levenshtein distance algorithm and have the effect of modifying the shape of a fuzzy set. If the result of a fuzzy operation reduces the membership value of part of an alignment to $< \beta$, that alignment will no longer be visualised or outputted. All fuzzy operations in FLAK are revocable.

3.1 Very Similar Alignments

Very is a concentration operation that narrows a set and reduces the degree of membership of fuzzy elements. The *very* operation is given as a mathematical square:

$\mu_A^{\text{Very}}(x) = [\mu_A(x)]^2$. A membership value of 0.82 in a fuzzy set *similar* will become 0.67 in the set of *very similar* alignments.

3.2 Extremely Similar Alignments

Extremely is also a concentration operation and is given by the 3rd power of a membership value: $\mu_A^{\text{Extremely}}(x) = [\mu_A(x)]^3$. A membership value of 0.82 in a fuzzy set *similar* will become 0.55 in the set of *extremely similar* alignments.

3.3 Very Very Similar Alignments

Very very is an extension of the basic *very* concentration and is computed as the square of the operation of concentration: $\mu_A^{\text{Very Very}}(x) = [\mu_A^{\text{Very}}(x)]^2 = [\mu_A(x)]^4$. A membership value of 0.82 in *similar* will become 0.45 in the set of *very very similar* alignments.

3.4 Fuzzy AND

A logical AND operation is the intersection ($\cap$) between two sets that contain shared elements. In fuzzy logic, this is expressed as $\mu_{A\cap B}(x) = min(\mu_A(x), \mu_B(x)) = \mu_A(x) \cap \mu_B(x)$ where $x \in X$. A fuzzy intersection is therefore the lower membership value in both sets of each element. FLAK implements AND by selecting the smallest $\mu_A(x)$ membership value from each chain of contiguous 32-mers that form an alignment.

3.5 Fuzzy OR

A logical OR can be implemented in fuzzy logic as the union ($\cup$) of two fuzzy sets and consists of every element that belongs to either set. The union is the largest membership value of the element in either set and is expressed as $\mu_{A\cup B}(x) = max(\mu_A(x), \mu_B(x)) = \mu_A(x) \cup \mu_B(x)$ where $x \in X$. The logical OR implementation in FLAK selects the largest $\mu_A(x)$ membership value from each chain of contiguous 32-mers in an alignment.

Table 2 Quality assessment of FLAK alignments using simulated genomes.

Size (Mbps)	# 32-mers	TP	FN	FP	Sn (%)	Sp (%)
1	31,250	31,248	2	1	99.99	99.99
10	312,500	312,498	2	1	99.99	99.99
50	1,562,500	1,562,496	4	5	99.99	99.99
100	3,125,000	3,124,998	2	24	99.99	99.99
250	7,812,500	7,812,500	0	47	100	99.99
500	15,625,000	15,624,996	4	187	99.99	99.99

3.6 Fuzzy NOT

The complement of a set is the logical NOT of a set and is expressed as $\mu_{\neg A}(x) = 1 - \mu_A(x)$. An alignment in FLAK with a $\mu_A(x)$ membership value of 0.75 will have a $\mu_{\neg A}(x)$ value of 0.25. If $\mu_{\neg A}(x) < \beta$, then FLAK will not display the alignment. Consequently, the fuzzy NOT operation works best at a low β-cutoff threshold and where the degree of similarity is weak.

4 Benchmarks and Results

This section describes the tests used to benchmark both the quality and running time of FLAK alignments. All results presented in this section were compiled from executing FLAK and the suite of representative whole-genome aligners on a LinuxMint 17.0.1 platform, with a 2.4 GHz Intel Core i7 processor and 16GB of RAM. An instance of the Java HotSpot 1.8 64-bit virtual machine was used as the runtime environment to test FLAK. The 13/20 spaced-seed ####-#-##-##-#-### and a β cut-off threshold of 0.85 was used for all tests.

4.1 Alignment Quality

To accurately measure and assess the quality of FLAK alignments, a set of synthetic query and references genomes were generated, ranging in size from 1Mbps – 500Mbps, along with a mapping file containing the indices of shared 32-mers between each query and reference sequence. After aligning with FLAK, both the sensitivity and specificity of the results were computed. The sensitivity (*Sn*) and specificity (*Sp*) of alignment was measured in the manner described by Nakato and Gotoh [2], i.e. $Sn = \frac{TP}{TP+FN}$ and $Sp = \frac{TP}{TP+FP}$ and are shown in Table 2.

Table 3 Comparison of alignment running times for both forward and reverse complements. Repeat filtering was not used for the prokaryotic genomes and the comparison of S.*pombe* and S.*cerevisiae*.

Query	Size (Mb)	Reference	Size (Mb)	FLAK T(s)	Nucmer T(s)	LAST T(s)	Cgaln T(s)
B.*suis* ATCC	1.9	B.*suis* 1330	2.1	1s	2s	14s	5s
N.*meningitidis* FAM	2.22	N.*meningitidis* MC	2.30	1s	7s	21s	15s
E.*coli* K12	4.7	E.*coli* 536	5.0	2s	7s	17s	9s
Y.*pestis* CO92	4.7	Y.*pestis* KIM	4.6	2s	11s	49s	14s
S.*pombe*	12.8	S.*cerevisiae*	12.2	5s	15s	52s	24s
P.*falciparum*	23.6	P.*yoelii*	22.4	63s	38s	4219s	>3hrs
A.*fumigatus*	29.8	A.*nidulans*	30.1	25s	43s	111s	177s
C.*elgans*	101.7	C.*briggsae*	92.5	82s	179s	988s	969s
D.*simulans*	119.9	D.*melanogaster*	122.1	155s	388s	1166s	1250s
D.*yakuba*	121.3	D.*melanogaster*	122.1	153s	532s	1778s	673s
D.*miranda*	138.7	D.*melanogaster*	122.1	154s	281s	1270s	447s
H.*sapiens* ChX	158.2	M.*musculus* ChX	169.2	304s	311s	9542s	759s
P.*abelii* Ch1	264.7	H.*sapiens* Ch1	231.1	502s	3395s	>3hrs	240s

4.2 Running Time Comparison

The running time of FLAK was benchmarked against a candidate set of pre-eminent whole-genome aligners. Mummer [1] was chosen as the existing *de facto* standard and because of its underlying suffix tree model. Both Cgaln [2] and LAST [3] are based on the *seed and extend* model, but use spaced-seeds to seed an alignment. In addition, LAST and Cgaln utilise the specific spaced-seeds recommended by Mak and Benson [11] that can also be used by FLAK. Both Nucmer and LAST were executed with their recommended default parameters. Cgaln requires the creation of a set of seed tables before comparing two sequences. For this benchmarking process, the *maketable* command was parametrised with the *-K13* switch to force the use of the same 13/20 spaced seed, 1111*1**11**11*1*111, used by FLAK. The *Cgaln* command was executed with the parameters *-r -ia -k3*. The *-ia* switch was turned off for genomes >100Mbps. The Drosophila alignments include the Muller Elements A-F for D.*melanogaster* and D.*simulans*. D.*yakuba* was comprised of Muller Elements A, B/C, C/B, D, E and F.

The results in Table 3 illustrate the high speed of FLAK for alignments across a range of different genome sizes. In particular, the running times of FLAK were consistent across genomes that are known to contain a large number of repetitive sequences, such as N.*meningitidis*, P.*falciparum*, and H.*sapiens* ChX.

5 Conclusion

FLAK provides users with a simple, graphical, wizard-based system for configuring, executing and analysing a whole-genome alignment. In contrast with existing approaches to WGA, FLAK supports native approximate k-mer matching, enabling a more detailed analysis of an alignment and the identification of putative homologous regions that existing approaches often miss. FLAK is the only WGA system that allows the application of fuzzy operators to genomic alignments and the *post hoc* filtering of results. The running times exhibited by FLAK out-perform all existing whole-genome aligners for prokaryotic and simple eukaryotic sequences and can match or exceed the speed of existing software for large mammalian chromosomes and genomes.

References

1. Kurtz, S., Phillippy, A., Delcher, A., Smoot, M., Shumway, M., Antonescu, C., Salzberg, S.: Versatile and open software for comparing large genomes. Genome biology **5**(2), R12 (2004)
2. Nakato, R., Gotoh, O.: Cgaln: fast and space-efficient whole-genome alignment. BMC bioinformatics **11**(1), 224 (2010)
3. Kielbasa, S., Wan, R., Sato, K., Horton, P., Frith, M.: Genome Research. Adaptive seeds tame genomic sequence comparison **21**(3), 487–493 (2011)
4. Uricaru, R., Michotey, C., Chiapello, H., Rivals, E.: YOC, A new strategy for pairwise alignment of collinear genomes. BMC Bioinformatics **16**(1), 111 (2015)
5. Torreno, O., Trelles, O.: Breaking the computational barriers of pairwise genome comparison. BMC Bioinformatics **16**(1) (2015)
6. Earl, D., Nguyen, N., Hickey, G., Harris, R.S., Fitzgerald, S., Beal, K., Seledtsov, I., Molodtsov, V., Raney, B.J., Clawson, H., Kim, J.: Alignathon: a competitive assessment of whole-genome alignment methods. Genome research **24**(12), 2077–2089 (2014)
7. Reddy, T., Thomas, A., Stamatis, D., Bertsch, J., Isbandi, M., Jansson, J., Mallajosyula, J., Pagani, I., Lobos, E., Kyrpides, N.: The Genomes OnLine Database (GOLD) v. 5: a metadata management system based on a four level (meta) genome project classification. Nucleic Acids Research **1**(11), 950 (2014)
8. Healy, J., Chambers, D.: Approximate k-mer matching using fuzzy hash maps. IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB) **1**(11), 258–264 (2014)
9. Levenshtein, V.: Binary codes capable of correcting deletions, insertions, and reversals. Soviet Physics **10**(8), 707–710 (1966)
10. Choi, K., Zeng, F., Zhang, L.: Good spaced seeds for homology search. In: Proceedings of the 4th IEEE Symposium on Bioinformatics (BIBE 2004), pp. 379–386 (2004)
11. Mak, D., Benson, G.: All hits all the time: parameter-free calculation of spaced seed sensitivity. Bioinformatics **25**(3), 302–308 (2009)
12. Ma, B., Tromp, J., Li, M.: PatternHunter: faster and more sensitive homology search. Bioinformatics **18**(3), 440–445 (2002)

Exploring the High Performance Computing-Enablement of a Suite of Gene-Knockout Based Genetic Engineering Applications

Zhenya Li, Richard O. Sinnott, Yee Wen Choon, Muhammad Farhan Sjaugi, Mohd Saberi Mohammad, Safaai Deris, Suhaimi Napis, Sigeru Omatu, Juan Manuel Corchado, Zuwairie Ibrahim and Zulkifli Md Yusof

Abstract Genetic engineering provides methods to modify the genes of microorganisms to achieve desired effects. This can be done for improved organism growth rate or increasing production yield of a desired gene product. Gene knockout is a

Z. Li · R.O. Sinnott
Department of Computing and Information Systems, University of Melbourne, Melbourne, Australia
e-mail: zhenyal@student.unimelb.edu.au, rsinnott@unimelb.edu.au

Y.W. Choon · M.S. Mohammad(✉)
Faculty of Computing, Universiti Teknologi Malaysia, Johor Bahru, Malaysia
e-mail: ewenchoon@gmail.com, saberi@utm.my

M.F. Sjaugi
School of Data Sciences, Perdana University, Serdang, Malaysia
e-mail: farhan@perdanauniversity.edu.my

S. Deris
Faculty of Creative Technology and Heritage, Universiti Malaysia Kelantan, Kota Bharu, Malaysia
e-mail: safaai@umk.edu.my

S. Napis
Faculty of Biotechnology and Biomolecular Sciences, Universiti Putra Malaysia, Serdang, Malaysia
e-mail: suhaimi@upm.my

S. Omatu
Department of Electronics, Information and Communication Engineering,
Osaka Institute of Technology, Osaka, Japan
e-mail: omatu@cs.osakafu-u.ac.jp

J.M. Corchado
Biomedical Research Institute of Salamanca / BISITE Research Group, University of Salamanca, Salamanca, Spain
e-mail: corchado@usal.es

Z. Ibrahim · Z.M. Yusof
Faculty of Electrical and Electronics, Faculty of Manufacturing Engineering,
Universiti Malaysia Pahang, Gambang, Malaysia
e-mail: zmdyusof@ump.edu.my, zuwairie@ump.edu.my

M.S. Mohamad et al. (eds.), *10th International Conference on PACBB*,
Advances in Intelligent Systems and Computing 477,
DOI: 10.1007/978-3-319-40126-3_14

technique that can improve the specific characteristics of microorganisms by disabling selected sets of genes. However, microorganisms are complex and predicting the effects of gene modification is difficult. Several algorithms have been proposed to support a range of gene knockout strategies, including BAFBA, BHFBA and DBFBA. In this paper, scaling these algorithms and methods to utilise High Performance Computing (HPC) resources have been explored. The applications have been parallelized on HPC and the scalability and performance of these approaches were explored and documented.

Keywords High performance computing · HPC · Gene knockout · Genetic engineering · Bees algorithm · Flux balance

1 Introduction

Genes of microorganism can be modified to improve specific properties of the strain of microorganism [1]. However the models of microorganisms are complex and contain various reactions and pathways that can interact with one another in ways that cannot always be predetermined. Predicting the effects of gene modifications can be computationally expensive due to the number of genes and their interaction possibilities. Three applications have been developed to explore these issues: a hybrid of Bees Algorithm and Flux Balance Analysis (BAFBA) [2], Bees Hill Flux Balance Algorithm Analysis (BHFBA) [3] and Differential Bees Flux Analysis (DBFBA) [4]. BAFBA, BHFBA and DBFBA are both aimed at calculating the optimization of gene knockout strategy to improve the desired effects with Bees algorithm used for searching optimized solution. These applications were developed to work on a single personal computer or workstation. Hence the scalability of these applications to exploit larger scale computing resources such as High Performance Computing (HPC) is highly desirable.

According to Valentini et al [5], a HPC represents a number of independent computers connected to each other through high-speed network. HPC as a way of providing reliable, scalable and available computing resources has attracted a lot of attention. With the development and availability of HPC, many applications can obtain benefit from this powerful computing resources and can gain improved performance through parallelisation and yet [unfinished sentence. And yet...??]. The availability and scalability provided by HPC is very attractive for applications that have a number of tasks that require minimal inter-tasks communication or independent tasks. This can be used to explore larger data sets or to run larger scales simulations. Although BAFBA, BHFBA and DBFBA applications are computationally intensive, these applications are trivially parallel and require no inter-process communication between tasks. Therefore, they are highly suited to the processing of multiple independent (concurrent) tasks. To make BAFBA, BHFBA and DBFBA applications to works on HPC platform, modification to the applications were made to make them fully utilize the performance of the HPC facility.

2 Parallelisation of Applications

BAFBA, BHFBA, DBFBA rely on interacting with the user to obtain their input tasks. Firstly, this aspect needs to be removed because user interaction is impractical for parallelisation. Instead, applications will obtain their input arguments from the program that calls them at runtime. Secondly, all three applications require similar input arguments. In order to work, all three applications require a SBML model representing the metabolic model, a list of reactions that will be considered, target reaction that is meant to be improved, substrate reactions and the maximum number of genes to be knocked out. BHFBA and DBFBA will also take an "imax" value, which defines the maximum limit of processing iterations. Since the input tasks are similar, a task script was created to handle the inputs. The task script can process the input arguments and call the corresponding program to handle that task (i.e. only one application shall be chosen in a single task). However users can list tasks for the three applications with similar input arguments in the same tasks list. This script is also responsible for setting the correct environment for the bioinformatics applications to run. With the modifications and the new script, users are allowed to input mixed tasks to the application and the tasks can subsequently be submitted to the cluster headnode and be processed concurrently by the cluster worker nodes. In order to efficiently process the tasks on the cluster, another script was written to distribute the input tasks among multiple CPUs through the Message-Passing Interface (MPI) library and subsequently collecting the outputs after each node has finished processing its job.

To distribute the tasks, two task-distributing methods were developed: Bulk and Master-Slave. In Bulk task-distributing method, large input tasks are divided into chucks evenly which are distributed to all available CPU cores, while in Master-Slave task-distributing method, one CPU core is designated as "Master" CPU core which would be responsible for fetching tasks from input task list and distributing the tasks to other CPU cores. The remaining CPU cores can request and process tasks from"Master" CPU core. In addition to Bulk task-distributing method, Bulk-R task distributing method was developed as an extension of the "Bulk" method, which will randomly permute the tasks in the task list. This method is designed to distribute the workload of tasks more evenly. The user is given the choice to select

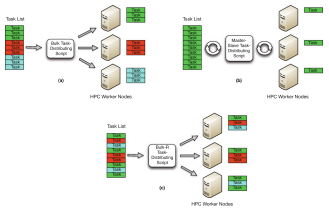


Fig. 1 (a) Bulk task-distributing method. (b) Master-Slave task-distributing method. (c) Bulk-R task-distributing method.

which task-distributing method fits with their projects. Figure 1 illustrates how the tasks are distributed to the HPC worker nodes for execution.

The main advantage of "porting" the three applications to HPC platform is so that multiple tasks can be executed concurrently at one time. Hence users will get the results faster than when executing the tasks on their workstation. This is very useful if the users want to do parameter studies of their metabolic models.

3 Performances Evaluation

Several experiments were completed on the University of Melbourne HPC cluster for benchmarking the performance of the two task distributing methods. Each node in the facility has 16 x 2.0GHz CPU cores. At one time, 1 CPU core can only process one task (but the same CPU core can take another task once the existing task has completed) and each node could execute up to 16 tasks at one time. The number of CPU cores per node was limited to 4, 8 and 16 CPU cores per node, hence the number of nodes ranged from 1 node to 4 nodes, depending on the limit. Two task lists were used as experimental inputs: tasklist128b and tasklist128nb. The tasklist128b input contained 128 similar tasks (one application with variety of parameters), i.e. the workload of each task was the same. The tasklist128nb contained a mix of tasks for all three applications. Because the tasks of BAFBA requires 100 iterations by default while the maximum iterations of tasks for BHFBA and DBFBA are all set to 10 in the list, the workload of tasks in tasklist128nb varies greatly.

Fig. 2 and 3 show the speedup of "bulk" and "master-slave" methods under different situations. As can be seen from Figure 2, the bulk method provides better performance when the workloads of input tasks are all similar. However, when the workload of tasks significantly varies, the master-slave method provides a better performance than the bulk method when the number of available cores exceeds a threshold as seen at Fig. 3. Besides the performance of the two methods, the impacts of file input size were also investigated. The input lists were then changed to 16, 64 and 128 number of tasks with mixed application tasks and the number of CPU cores were fixed to 8. Fig. 4 demonstrates the speedup of two methods when the input size was changed. The speedup of the Bulk method has a positive relationship with the input size and eventually exceeds the value of Master-Slave. On the other hand, the performance of Master-Slave is more stable with regard to the change of input size. Hence, users are advised to use Master-Slave method when the availabity of the CPU core for execution is limited.

The output of these applications includes a binary file, which contains the result and two graph images. The result includes the list of reactions that should be knocked out, the genes that are related with the knocked out reactions, the predicted BPCY (Biomass Product Coupled Yield) and the predicted growth rate. For validating the performance of the applications on HPC cluster, i.e. comparing with the original application, the task "E coli iJR904-BHFBA-all-UMPK-UNK3-5-10" was executed on both a local machine and the cluster. The Matlab file that was generated by the local

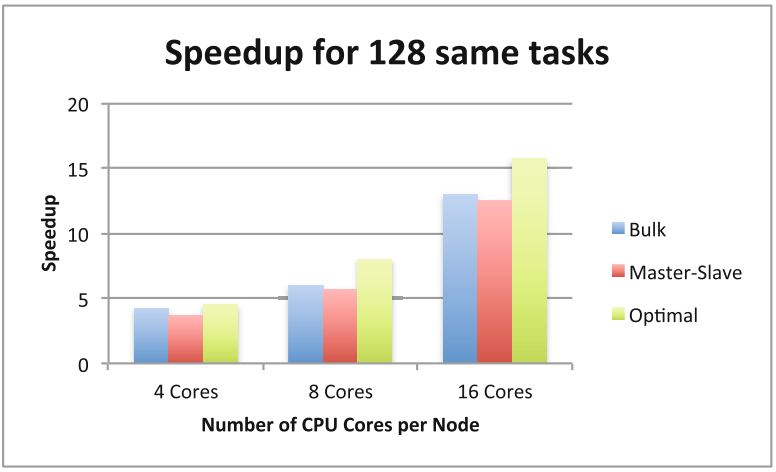


Fig. 2 Speedup of Bulk and Master-Slave methods with 128 same tasks

machine and the cluster are similar. Both outputs identified that 'AMPN', 'ATPM', 'HDCAt2', 'MI1PP' and 'PUNP2' are reactions that should be knocked out and the predicted values of BPCY and growth rate were calculated as 40.2889 and 0.95697 respectively.

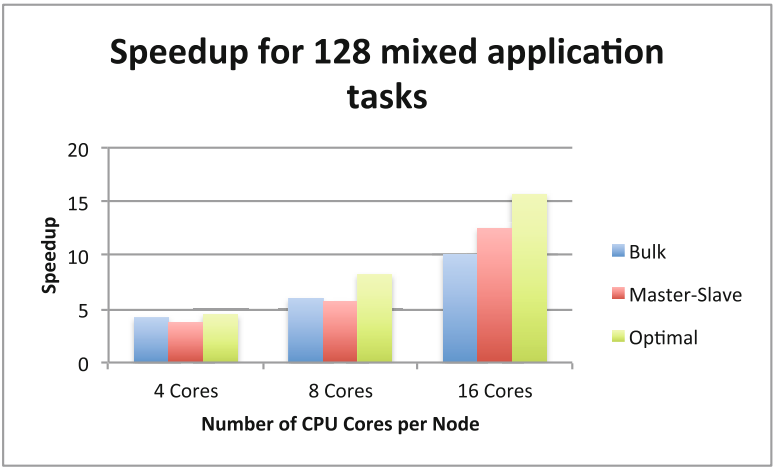


Fig. 3 Speedup of Bulk and Master-Slave methods with 128 mixed tasks

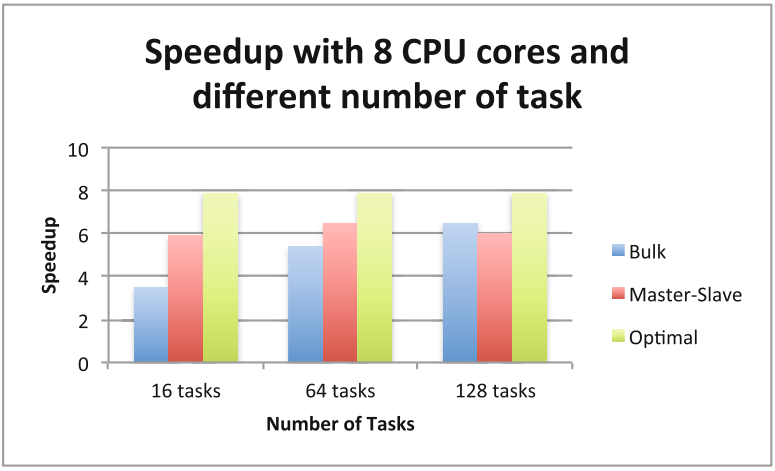


Fig. 4 Speedup of different input size

4 Conclusion

HPC resource can provide large scale computing power. However to exploit such facility, applications often need modifications to utilize computing resources efficiently. This project focused on providing tools to reduce the difficulties of exporting applications to HPC. The BAFBA, BHFBA and DBFBA applications were ported to use the HPC cluster. This method can be generalized to handle similar projects that need to process many independent tasks. Two types of task distributing methods were provided for users to choose from. The Master-Slave method provides better performance when the input tasks have various levels of workload with limited number of CPU core due to its near constant speedup. However when CPU cores have increased availability, the Bulk method is generally a better option. The performance of the Bulk method also increases when the number of CPU core is increased. Futhermore larger input sizes can bring better performance for the bulk method, therefore users are suggested to export their applications to the cluster when the input tasks requires a lot of processing time. However the threshold of the input size for HPC resources needs further investigation and the threshold value might rely on specific properties of the application and the input tasks. To submit the tasks to the HPC cluster, users currently shall use command line interface. However some users may have difficulty due to lack of knowledge with Linux command line interface. Hence for future work, a web interface could also be developed to simplify the user interaction.

Acknowledgments We would like to thank Universiti Teknologi Malaysia for funding this research through the Research University Grant (Grant number: Q.J130000.2528.12H12). This research is also supported by the Malaysian Ministry of Higher Education through the Fundamental Research Grant Scheme (Grant number: R.J130000.7828.4F720 and RDU140114).

References

1. Burgard, A.P., Pharkya, P., Maranas, C.D.: Optknock: a bilevel programming framework for identifying gene knockout strategies for microbial strain optimization. Biotechnology and bioengineering **84**(6), 647–657 (2003)
2. Choon, Y.W., Mohamad, M.S., Deris, S., Illias, R.M., Chong, C.K., Chai, L.E.: A hybrid of bees algorithm and flux balance analysis with OptKnock as a platform for in silico optimization of microbial strains. Bioprocess and biosystems engineering **37**(3), 521–532 (2014)
3. Choon, Y.W., Mohamad, M.S., Deris, S., Chong, C.K., Omatu, S., Corchado, J.M..: Gene Knockout Identification Using an Extension of Bees Hill Flux Balance Analysis. BioMed research international (2015)
4. Choon, Y.W., Mohamad, M.S., Deris, S., Illias, R.M., Chong, C.K., Chai, L.E., Omatu, S., Corchado, J.M.: Differential Bees Flux Balance Analysis with OptKnock for *In Silico* Microbial Strains Optimization. PLoS ONE **9**(7), e102744 (2014)
5. Valentini, G.L., Lassonde, W., Khan, S.U., Min-Allah, N., Madani, S.A., Li, J., Zhang, L., Wang, L., Ghani, N., Kolodziej, J., Li, H., Zomaya, A.Y., Xu, C.-Z., Balaji, P., Vishnu, A., Pinel, F., Pecero, J.E., Kliazovich, D., Bouvry, P.: An overview of energy efficiency techniques in cluster computing systems. Cluster Computing **16**(1), 3–15 (2013)

RUbioSeq+: An Application that Executes Parallelized Pipelines to Analyse Next-Generation Sequencing Data

Miriam Rubio-Camarillo, Hugo López-Fernández, Gonzalo Gómez-López, Ángel Carro, José María Fernández, Florentino Fdez-Riverola, Daniel Glez-Peña and David G. Pisano

Abstract To facilitate routine analysis and to improve the reproducibility of the results, next-generation sequencing analysis requires intuitive, efficient and integrated data processing pipelines. Here, we present RUbioSeq+, a multi-platform application that incorporates a suite of automated and parallelized workflows to analyse NGS data. The software supports DNA-seq (single-nucleotide and copy number variation analyses) as well as for bisulfite-seq and ChIP-seq workflows. RUbioSeq+ supports parallelized and multithreaded execution, and its interactive graphical user interface facilitates its use by both biomedical researchers and bioinformaticians. Results generated by our software have been experimentally validated and accepted for publication. RUbioSeq+ is free and open to all users at http://rubioseq.bioinfo.cnio.es/.

M. Rubio-Camarillo G. Gómez-López Á. Carro D.G. Pisano
Bioinformatics Unit (UBio), Structural Biology and Biocomputing Programme,
Spanish National Cancer Research Centre (CNIO), Madrid, Spain
e-mail: {mrubioc,ggomez,acarro,dgonzalez}@cnio.es

H. López-Fernández F. Fdez-Riverola D. Glez-Peña(✉)
ESEI - Escuela Superior de Ingeniería Informática, Edificio Politécnico,
Campus Universitario As Lagoas s/n, Universidad de Vigo, 32004 Ourense, Spain
e-mail: {hlfernandez,riverola,dgpena}@uvigo.es

H. López-Fernández F. Fdez-Riverola D. Glez-Peña
Instituto de Investigación Biomédica de Vigo (IBIV), Vigo, Spain

J. María Fernández
Structural Computational Biology Group, Structural Biology and BioComputing Programme, Spanish National Cancer Research Centre (CNIO),
3rd Melchor Fernández Almagro St., 28029 Madrid, Spain
e-mail: jmfernandez@cnio.es

M.S. Mohamad et al. (eds.), *10th International Conference on PACBB*,
Advances in Intelligent Systems and Computing 477,
DOI: 10.1007/978-3-319-40126-3_15

Keywords NGS analysis · Parallelized workflows · Whole-genome · Variant calling · ChIPSeq · Bisulfite-Seq · CNV · HPC · SGE

1 Introduction

The increasing use of next-generation sequencing (NGS) studies has revealed the need for integrated and reliable pipelines to analyse deep-sequencing experiments in a reproducible way. This issue is especially relevant in hospitals and research institutes where regular analyses accentuates the demand for intuitive and automated workflows that accelerate the delivery of final results, minimizing human technical error, and ensuring the reproducibility and fidelity of the data obtained.

NGS data is usually analysed in a set of successive stages that are executed routinely. In this scenario there are highly specific software available to carry out each of these particular steps, constituting a growing and diverse catalogue including quality control utilities, read aligners, variant callers, peak finders, functional annotators, mutational impact predictors, etc. (1,2). In such a diverse scenario, the quality of all these tools is very heterogeneous in terms of implementation and documentation, and in many cases the applications could be difficult to understand for non-specialist users, requiring a solid expertise in bioinformatics to manage them properly.

In the light of this situation, a number of initiatives have been proposed to provide effective solutions that facilitate the systematic analysis of NGS experiments. For example, HugeSeq (3), Bcbio-nextgen (4) and Omics-pipe (5) are recent open-source pipelines available to analyse NGS data in an automated and proficient manner. These examples reflect the remarkable effort to provide powerful specialized computational frameworks to analyse NGS data. However, they usually lack an adequate Graphic User Interface (GUI) and thus, they are cumbersome for researchers without computational or bioinformatics skills. Besides, current version of HugeSeq and Bcbio-nextgen do not support the analysis of several seq-based experiments, such as copy-number alteration sequencing (CNA-seq), bisulfite-seq or ChIPseq. Galaxy (6) represents a large and flexible web-based platform that provides support for NGS experiments (e.g. ChIPseq, variant analysis, etc.) and although it may be installed as a local server, its set up requires advanced technical skills. Despite its potential, Galaxy's NGS toolbox is still in beta state and it does not support either CNA-seq or bisulfite-seq analysis. Moreover, its online use demands high bandwidth capacities due to the size of the NGS files. In addition, other interesting proposals such as GeneProf (7) or CisGenome (8) do perform ChIPseq analysis through an interactive GUI, but they do not provide support for other types of NGS techniques.

Here we present RUbioSeq+, a multi-platform application for the integrated analysis of NGS data. This software uses well-established tools to implement pipelines for DNA-seq, CNA-seq, bisulfite-seq and ChIP-seq experiments. RUbioSeq+ is free and includes the entire core functionalities implemented in the

original release of RUbioSeq (9) while expanding its capabilities by supporting parallelized analysis of full genomes in computing farms. Moreover, we have included two novel pipelines for ChIPseq analysis covering (i) sample quality control, read alignments, assessment of biological replicates, and (ii) peak calling for the detection of histone marks and transcription factors binding sites. RUbioSeq+ also incorporates a new and user-friendly GUI, designed for interdisciplinary research groups where bioinformaticians and biomedical researchers work together.

2 Implementation

RUbioSeq+ is written in Perl language. It is designed to run on ordinary UNIX workstations and it has been successfully tested in Linux and Mac OS X, as well as on HPC systems with SGE or PBS as cluster job schedulers. Windows users may also run RUbioSeq+ using the Docker client (http://www.docker.com). Its modular programming design provides a high degree of flexibility to facilitate the creation of shared libraries and functions through the distinct execution branches (e.g., the cluster job management module) that stabilizes the code, helps with software maintenance and avoids code redundancy. This facet will also facilitate the inclusion of additional functionalities and extensions in future versions of the tool.

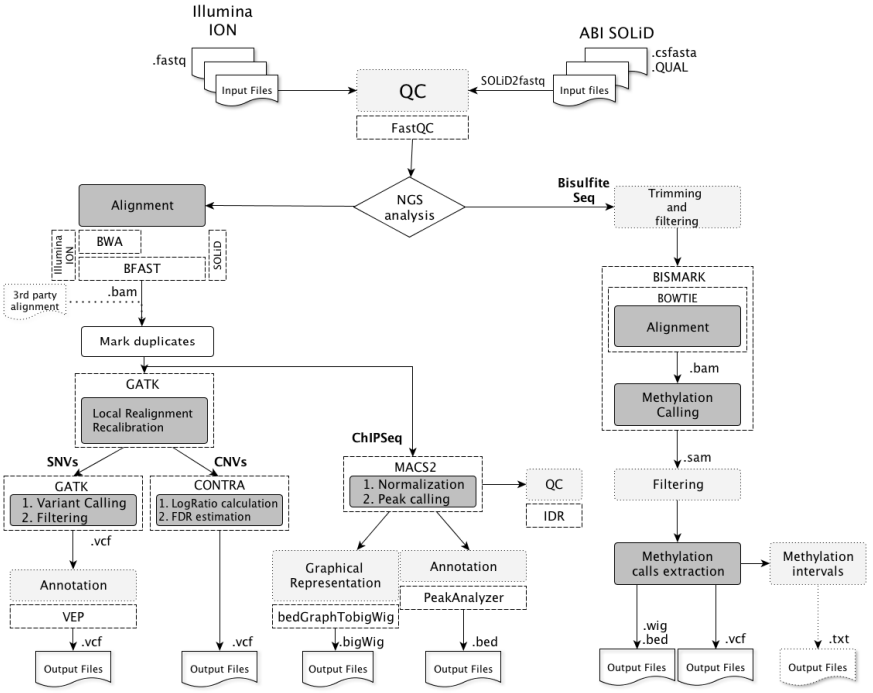


Fig. 1 Schematic RUbioSeq+ flow diagram.

Our software employs standard formats of input and output files, supporting both single and paired-end experiments. The execution of each pipeline starts from either raw data files (FASTQ and ABI SOLiD formats) or alignments (BAM, see Figure 1), after which RUbioSeq+ is very flexible and adjustable, allowing the users to customize the parameters employed at each stage of the workflow. In order to assure the reproducibility of the workflows employed, our software automatically exports the full technical configuration of the pipeline into a .XML file. When required, this configuration file can be reloaded into RUbioSeq+, replicating the technical conditions used in the original execution. By reusing these configuration files, every analysis performed by RUbioSeq+ may be controlled, evaluated and systematically reproduced, provided that the same version of RUbioSeq+ is used.

ChIPseq

RUbioSeq+ provides two novel workflows to analyze ChIPseq experiments. Using FASTQ or BAM files as primary input, RUbioSeq+ can use MACS2 (10) and CCAT (11) to detect sharp or broad peaks, respectively.

The main steps in the workflows available include: a) Quality control through FastQC (12); b) short-read alignment with BWA (13); c) duplicate marking with Picard tools (14); d) normalization performed with SAMtools (15) in which the files are equilibrated in terms of size to make the experiments comparable; e) peak calling with MACS2 or CCAT; f) an optional step to assess the reproducibility of biological replicates using IDR (16); and f) peak annotation with PeakAnnotator (17).

In both the MACS2 and CCAT workflows, peak calling steps are performed on each sample. This strategy allows experiments to be analysed in parallel, making the protocol adaptable to support a variety of experimental ChIPseq designs (e.g. case+input, case versus control, case+input vs control+input, etc.).

DNA-seq

This workflow accepts FASTQ, ABI SOLID raw data or BAM files as the primary input to analyse single-nucleotide variants in full genomes and exomes. The pipeline is divided into four main modules: (a) short-read alignment with BWA and BFAST (18), and a quality control analysis using FastQC; (b) duplicate marking using Picard tools, realignment and recalibration using GATK (19, 20); (c) GATK variant calling (variant standard database annotation) and advanced filtering permitting the user to choose between Hard Filtering using GATK's VariantFiltration walker or GATK's variant quality score recalibrator (VQSR); and (d) Pair-wise comparisons to detect variants (e.g. case versus control). Variant impact is calculated using Ensembl Variant Effect Predictor (VEP, 21).

CNA-seq

RUbioSeq+ performs CNV detection for exome sequencing experiments using FASTQ and BAM as the input files. This workflow uses: (a) BWA and BFAST aligners to generate BAM files; (b) GATK to mark duplicates and for alignment recalibration; and (c) CONTRA software (22) to estimate case-control log ratios for each region targeted, and to evaluate the false discovery rate (FDR) of the gains and losses detected.

Bisulfite-seq

RUbioSeq+ integrates a bisulfite-seq workflow that supports the analysis of full genomes from FASTQ files. The workflow extracts methylation calls for each independent sample included in a particular experiment, but it can also construct a unique file with the whole calling information from the full set of samples included in such experiments. The pipeline includes: (a) quality control, sequence alignment and methylation calling with Bismark (23); (b) methylation call extraction; and (c) an optional calculation of the percentage methylation in specific intervals.

In light of the overwhelming list of applications currently available for NGS data analysis, we have selected well-established software to construct workflows for RUbioSeq+ (Table 1). The workflows involve operative modules acting at different functional levels, each level representing an independent stage of the analysis (e.g. alignment level, calling level). The level-based design allows users to launch the workflows in their entirety or partially, depending on the laboratory requirements, thereby providing the greatest flexibility and speed when running the analyses using different parameters. For example, the ChIPseq workflow could be executed at four levels depending on the input stage: 1) alignment phase and FastQC analysis if the user starts with unaligned reads; 2) duplicate markings if the reads have been aligned but the user wants to check the level of library duplication; 3) normalization steps if the libraries to be compared have to be balanced; and/or 4) straightforward ChIPseq calling, peak annotation and IDR control if allowed by the input data. Once the workflow has been fully executed, the users might want to modify the parameters of the peak caller and rerun the analysis from step 3, without having to execute the full workflow. Each round of execution will generate the associated result files, together with their corresponding process logs for identification.

Table 1 Software included in RubioSeq+

Shared Software		**Branch-specific software**		
Java	1.7	**SNV**	VEP	73
Samtools	0.1.19	**CNV**	CONTRA	2.0.3
PicardTools	1.107	**ChipSeq**	MACS2	2.0.10
FastQC	0.10.1		CCAT	3
BWA	0.7.10		IDR	1
Bfast-bwa	0.7.0b	**Methylation**	Bismark	0.10.1
VCFtools	Not used		Bowtie	0.12.7
BEDTools	2.16.2		Fastx Toolkit	0.0.13.2
GATK	3.1-1		FiloTools	1.1.0

To adapt RUbioSeq+ to the analysis of whole genomes, we reimplemented the alignment step using a parallelized design. Thus, starting from the fragmented raw data files supplied by sequencers, RUbioSeq+ processes the fragments side-by-side to generate a single alignment file that can be used as an input in the next stage. This implementation saves time and computational demands, performing the analysis of whole genomes in an efficient manner.

The RUbioSeq+ GUI is implemented as an AJAX-enabled web application programmed in Java 1.7. The ZK development framework was used to construct a rich web user interface with many features of a desktop application, and a HSQLDB (HyperSQL DataBase) is used to store the application data (users, configurations, experiments, etc.). Since the user interface has been developed using web technologies, the RUbioSeq+GUI can be used in two different modes: (a) as a stand-alone application using Jetty Runner 8.1; or (b) as a web front-end that is installed in a dedicated web server. The second option is especially useful for users of a bioinformatics unit that have at their disposal a computer cluster or dedicated server. In such a scenario, RUbioSeq+ can be installed into the head node of the cluster so that users can access RubioSeq+GUI via the server, configuring their experiments there and launching analyses that will be executed by the cluster machines (with SGE or PBS as cluster job schedulers).

3 Results and Discussion

We present here RUbioSeq+, a novel and improved version of the RUbioSeq suite for the analysis of NGS data. Our application consists of a multiplatform collection of automated and parallelized pipelines to analyse DNA-seq, CNA-seq and bisulfite-seq experiments. Additionally, RUbioSeq+ includes two new ChIP-seq workflows based on MACS2 and CCAT tools for peak calling. The software can provide an intuitive GUI that can be implemented to support integrated workflows for bioinformaticians and biomedical researchers, and it is being used extensively at our institute for both research and clinical purposes.

RUbioSeq+ workflows are divided into distinct analytical branches that may be executed independently to analyse distinct NGS experiments. Moreover, the default parameters have been tweaked based on developer's recommendations and on our daily experience performing routine NGS analyses for biomedical researchers.

Our application automates the NGS analysis, reducing human errors and improving the reproducibility of deep-sequencing studies. To this aim, the software generates an XML configuration file that is associated to every execution and that may be imported to reproduce the same technical conditions of any particular implementation. Additionally, RUbioSeq+ provides an intuitive framework in which each pipeline can be checked and controlled. Thus, quality control of the sample is addressed in every workflow, and all the processes are monitored through log files to trail the progress and errors at both the sample and

data analysis levels. The complete results of RUbioSeq+ are saved in a project directory tree that maintains a structured organization for the output files.

Since legal requirements impede many laboratory computers holding sensitive data from being connected to the internet, RUbioSeq+ can also be run on a computer that is offline. In this particular case, some of the online functionalities required by RUbioSeq+ (e.g., VEP) must be installed locally onto the computer. Accordingly, RUbioSeq+'s administrator would have to set up VEP to run on the local installation.

The RUbioSeq+ GUI also facilitates the administration of the tasks it carries out and thus, managing the technical configuration of RUbioSeq+ is straightforward when handled through the administrator's profile. In addition, the RUbioSeq+ web site offers exhaustive help, documentation and video tutorials (http://rubioseq.bioinfo.cnio.es).

Finally, despite depending on more than 20 different software packages, some of them difficult to install and setup, the installation of RUbioSeq+ is a straightforward process. The software is easy to configure and examples of its use are provided as step-by-step video tutorials. In addition, three installation options are provided: 1) a customized 64-bit LiveDVD based on Ubuntu 14.04.1 on which RUbioSeq+ and all its dependencies are bundled, ready to be used on any computer; 2) A Docker image (ubio/rubioseq:latest) stored in the public Docker Hub; and 3) Manual installation through a simple script provided with the software.

4 Conclusions

RUbioSeq+ is a multiplatform application for the integrated analysis of NGS data. Our software implements pipelines for the analysis of single nucleotide, copy-number variation, bisulfite-seq and ChIP-seq experiments using well-established tools to perform these common tasks. The results obtained by RUbioSeq+ have already been validated and published (24, 25). The RUbioSeq+ GUI developed by the SING group is licensed under a GNU GPL 3.0 License (http://www.gnu.org/copyleft/gpl.html). Moreover, the RUbioSeq+ software and full documentation are free and publicly available under Creative Commons License at http://rubioseq.bioinfo.cnio.es

Acknowledgements M.R-C. is funded by the BLUEPRINT Consortium (FP7/ 2007-2013) under grant agreement number 282510. J.M.F is funded by the Spanish National Institute of Bioinformatics (INB), a project financed by the Spanish Ministry of Economy and Competitiveness (BIO2007-666855). H.L-F is funded by a predoctoral fellowship from the Xunta de Galicia. In addition, this work was partially funded by the [14VI05] Contract Programme of the University of Vigo. F.F-R and D.G-P are funded by the European Union's Seventh Framework Programme FP7/REGPOT 2012 2013.1 under grant agreement n° 316265 (BIOCAPS), the "Agrupamento INBIOMED" from DXPCTSUG-FEDER "unha

maneira de facer Europa" (2012/273) and the "Platform of integration of intelligent techniques for analysis of biomedical information" project (TIN2013-47153-C3-3-R) financed by the Spanish Ministry of Economy and Competitiveness. We are thankful to CNIO's Translational Bioinformatics Unit staff for their assistance in beta testing and for their useful suggestions. We also thank E. Carrillo-de-Santa-Pau and F. Al-Shahrour for their advice regarding the manuscript and for fruitful discussions.

References

1. Trapnell, C., Salzberg, S.L.: How to map billions of short reads onto genomes. Nat. Biotech. **27**(5), 455–457 (2009)
2. Ding, L., Wendl, M.C., McMichael, J.F., Raphael, B.J.: Expanding the computational toolbox for mining cancer genomes. Nat. Rev. Genet. **15**(8), 556–570 (2014)
3. Lam, H.Y., Pan, C., Clark, M.J., Lacroute, P., Chen, R., Haraksingh, R., O'Huallachain, M., Gerstein, M.B., Kidd, J.M., Bustamante, C.D., Snyder, M.: Detecting and annotating genetic variations using the HugeSeq pipeline. Nat. Biotechnol. **30**(3), 226–229 (2012)
4. bcbio toolkit. https://bcbio-nextgen.readthedocs.org
5. Omics Pipe. https://bitbucket.org/sulab/omics_pipe
6. Goecks, J., Nekrutenko, A., Taylor, J., The Galaxy Team: Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences. Genome Biol. **11**(8), R86 (2010)
7. Halbritter, F., Vaidya, H.J., Tomlinson, S.R.: GeneProf: analysis of high-throughput sequencing experiments. Nat. Methods **9**(1), 7–8 (2011)
8. Ji, H., Jiang, H., Ma, W., Johnson, D.S., Myers, R.M., Wong, W.H.: An integrated software system for analyzing ChIP-chip and ChIP-seq data. Nat. Biotechnol. **26**(11), 1293–1300 (2008)
9. Rubio-Camarillo, M., Gómez-López, G., Fernández, J.M., Valencia, A., Pisano, D.G.: RUbioSeq: a suite of parallelized pipelines to automate exome variation and bisulfite-seq analyses. Bioinformatics **29**(13), 1687–1689 (2013)
10. Zhang, Y., Liu, T., Meyer, C.A., Eeckhoute, J., Johnson, D.S., Bernstein, B.E., Nusbaum, C., Myers, R.M., Brown, M., Li, W., Liu, X.S.: Model-based Analysis of ChIP-Seq (MACS). Genome Biology **9**, R137 (2008)
11. Xu, H., Handoko, L., Wei, X., Ye, C., Sheng, J., Wei, C.L., Lin, F., Sung, W.K.: A signal-noise model for significance analysis of ChIP-seq with negative control. Bioinformatics **26**(9), 1199–1204 (2010)
12. FastQC project. http://www.bioinformatics.babraham.ac.uk/projects/fastqc/
13. Li, H., Durbin, R.: Fast and accurate short read alignment with Burrows-Wheeler transform. Bioinformatics **25**(14), 1754–1760 (2009)
14. Picard tools. http://broadinstitute.github.io/picard/
15. Li, H., Handsaker, B., Wysoker, A., Fennell, T., Ruan, J., Homer, N., Marth, G., Abecasis, G., Durbin, R., 1000 Genome Project Data Processing Subgroup: The Sequence alignment/map (SAM) format and SAMtools. Bioinformatics **25**, 2078–2079 (2009)
16. Li, Q., Brown, J., Huang, H., Bickel, P.: Measuring reproducibility of high-throughput experiments. Ann. Appl. Stat. **5**, 1752–1779 (2011)

17. Salmon-Divon, M., Dvinge, H., Tammoja, K., Bertone, P.: PeakAnalyzer: Genome-wide annotation of chromatin binding and modification loci. BMC Bioinformatics **11**, 415 (2010)
18. Homer, N., Merriman, B., Nelson, S.F.: BFAST: an alignment tool for large scale genome resequencing. PLoS One **4**(11), e7767 (2009)
19. DePristo, M., Banks, E., Poplin, R., Garimella, K., Maguire, J., Hartl, C., Philippakis, A., del Angel, G., Rivas, M.A., Hanna, M., McKenna, A., Fennell, T., Kernytsky, A., Sivachenko, A., Cibulskis, K., Gabriel, S., Altshuler, D., Daly, M.: A framework for variation discovery and genotyping using next-generation DNA sequencing data. Nature Genetics **43**, 491–498 (2011)
20. Van der Auwera, G.A., Carneiro, M., Hartl, C., Poplin, R., del Angel, G., Levy-Moonshine, A., Jordan, T., Shakir, K., Roazen, D., Thibault, J., Banks, E., Garimella, K., Altshuler, D., Gabriel, S., DePristo, M.: From FastQ Data to High-Confidence Variant Calls: The Genome Analysis Toolkit Best Practices Pipeline. Current Protocols in Bioinformatics **43**, 11.10.1–11.10.33 (2013)
21. McLaren, W., Pritchard, P., Rios, D., Chen, Y., Flicek, P., Cunningham, F.: Deriving the consequences of genomic variants with the Ensembl API and SNP Effect Predictor. Bioinformatics **26**(16), 2069–2070 (2010)
22. Li, J., Lupat, R., Amarasinghe, K.C., Thompson, E.R., Doyle, M.A., Ryland, G.L., Tothill, R.W., Halgamuge, S.K., Campbell, I.G., Gorringe, K.L.: CONTRA: copy number analysis for targeted resequencing. Bioinformatics **28**(10), 1307–1313 (2012)
23. Krueger, F., Andrews, S.R.: Bismark: a flexible aligner and methylation caller for Bisulfite-Seq applications. Bioinformatics **27**(11), 1571–1572 (2011)
24. Vaqué, J.P., Gómez-López, G., Monsálvez, V., Varela, I., Martínez, N., Pérez, C., Domínguez, O., Graña, O., et al.: PLCG1 mutations in cutaneous T-cell lymphomas. Blood **123**(13), 2034–2043 (2014)
25. Cuadrado, A., Remeseiro, S., Graña, O., Pisano, D.G., Losada, A.: The contribution of cohesin-SA1 to gene expression and chromatin architecture in two murine tissues. Nucleic Acids Res., March 3, 2015. doi:10.1093/nar/gkv144

Exceptional Symmetry Profile: A Genomic Word Analysis

Vera Afreixo, João M.O.S. Rodrigues, Carlos A.C. Bastos and Raquel M. Silva

Abstract The extension of Chargaff's second rule, also called the DNA symmetry, is pointed as an universal law present in the genomes of species. Previously, a measure of the symmetry above that expected in independence contexts (exceptional symmetry) was proposed to evaluate the phenomenon globally. The global exceptional symmetry was found in several species. However, the analysis of exceptional symmetry by word was not studied in detail. In this work a new exceptional symmetry measure is proposed to evaluate the exceptional symmetry effect by symmetric word pair. We develop a detailed study of the exceptional symmetry by symmetric pairs. We also discuss the exceptional symmetry by symmetric word pair for several organisms: 7 viruses; 5 archaea; 5 bacteria; 14 eukaryotes.

Keywords Chargaff's second parity rule · Exceptional symmetry · Genomic word counts

1 Introduction

Chargaff's first parity rule states that, in any sequence of double-stranded DNA molecules, the total number of complementary nucleotides is exactly equal [5].

V. Afreixo(✉) · R.M. Silva
iBiMED-Institute of Biomedicine, University of Aveiro, Campus Universitário de Santiago, Aveiro, Portugal
e-mail: vera@ua.pt

V. Afreixo · J.M.O.S. Rodrigues · C.A.C. Bastos · R.M. Silva
IEETA-Institute of Electronic Engineering and Informatics of Aveiro, University of Aveiro, Campus Universitário de Santiago, Aveiro, Portugal

V. Afreixo
Department of Mathematics, University of Aveiro, Campus Universitário de Santiago, Aveiro, Portugal

J.M.O.S. Rodrigues · C.A.C. Bastos
Department of Electronics, Telecommunications and Informatics, University of Aveiro, Campus Universitário de Santiago, Aveiro, Portugal

M.S. Mohamad et al. (eds.), *10th International Conference on PACBB*,
Advances in Intelligent Systems and Computing 477,
DOI: 10.1007/978-3-319-40126-3_16

Chargaff's second parity rule states that those quantities are almost equal in a single strand of DNA [8, 10, 11], and this phenomenon holds in almost all living organisms.

The extension to the second parity rule is also known as single strand symmetry phenomenon. The single strand symmetry states that, in each DNA strand, the proportion of an oligonucleotide should be similar to that of its reversed complement [6]. There is no knowledge about why the parity is needed in the DNA sequence and there is no consensual explanation for the occurrence of the single strand phenomenon. There are some attempts to explain the phenomenon related with the species evolution process, for example: stem-loops hypothesis [7]; duplication followed by inversion hypothesis [4]; inversions and inverted transpositions hypothesis [3].

Powdel and others [9] studied the symmetry phenomenon in non-overlapping regions of DNA of a specific size. They analysed the frequency distributions of the local abundance of oligonucleotides along a single strand of DNA, and found that the frequency distributions of reverse complementary oligonucleotides tends to be statistically similar. Afreixo et al. [2] introduced a new symmetry measure, which emphasizes that the frequency of an oligonucleotide is more similar to the frequency of its reversed complement than to the frequencies of other equivalent composition oligonucleotides. They also identified several word groups with a strong exceptional symmetry.

Based on the exceptional symmetry concept, we discuss this effect size measure for each symmetric word pairs, and we also propose a new measure to evaluate the dissimilarity between word occurrences in relation to the word dissimilarities in the corresponding equivalent composition group. We obtain the word symmetry effects for 31 complete genomes. Our results show that the symmetry effect value has the potential to discriminate between species groups. And there are sets of words which present high symmetry effect in all species under study, and we also indicate several species specificities.

2 Methods

In this study, we used the complete DNA sequences of 31 organisms obtained from the National Center for Biotechnology Information (NCBI; ftp://ftp.ncbi.nih.gov/genomes/). The species used in this work are listed in Tab. 1. The study was carried out in representative species of the major taxonomic groups across the tree of life and includes genomes from vertebrates, invertebrates, protozoans, fungi, plants, bacteria (gram-positive and gram-negative), archaea and virus (DNA and RNA viruses). The study could be extended to other organisms to improve the resolution of the species tree, but additional genomes will eventually become redundant.

All genome sequences used under this study were processed to obtain the word counts, considering overlap between successive words. We obtained the word counts for word lengths from 1 to 12 nucleotides.

We proposed in a previous work [2] the exceptional genomic word symmetry for equivalent composition groups (ECG) and globally. Some words are equal to

Table 1 List of organisms whose DNA was used in this study.

Organism	Group	Abbreviation
Homo sapiens[a]	eucarya (animalia)	HSap
Macaca mulatta[a]	eucarya (animalia)	MacM
Pan trogolodytes[a]	eucarya (animalia)	PanT
Mus musculus[a]	eucarya (animalia)	MusM
Rattus norvegicus[a]	eucarya (animalia)	RatN
Danio rerio[a]	eucarya (animalia)	DRer
Apis mellifera[a]	eucarya (animalia)	Apis
Caenorhabditis elegans[b]	eucarya (animalia)	CaeE
Drosophila melanogaster[b]	eucarya (animalia)	DroM
Arabidopsis thaliana[a]	eucarya (plantae)	AraT
Vitis vinifera[a]	eucarya (plantae)	VitV
Saccharomyces cerevisiae[a]	eucarya (fungi)	SacC
Candida albicans[a]	eucarya (fungi)	CanA
Plasmodium falciparum[a]	eucarya (protozoa)	PlaF
Helicobacter pylori[a]	bacteria	HelP
Streptococcus mutans GS[a]	bacteria	StMG
Streptococcus mutans LJ23[a]	bacteria	StML
Streptococcus pneumoniae[a]	bacteria	StPn
Escherichia coli[a]	bacteria	EscC
Aeropyrum camini[a]	archaea	AerC
Aeropyrum pernix[a]	archaea	AerP
Caldisphaera lagunensis[a]	archaea	CalL
Candidatus Korarchaeum[a]	archaea	CanK
Nanoarchaeum equitans[a]	archaea	NanE
NC001341[a]	virus	AbaS
NC001447[a]	virus	AcaT
NC004290[a]	virus	AchD
NC008724[a]	virus	AcPL
NC011646[a]	virus	AcPM
NC011591[b]	virus	SouT
NC012532[b]	virus	ZikV

[a] Downloaded in January 2014. [b] Downloaded in March 2016.

their reversed complement, we denote these as self symmetric words (SSW). We also define a symmetric word pair as the set composed by one word w and the corresponding reversed complement word w', with $(w')' = w$ (for example, CCA and TGG make a symmetric word pair). Let n_w be the total number of occurrences of word w in the sequence, n_m be the total number of occurrences of words in the ECG G_m which contain words composed by m nucleotides A or T.

The measure R was proposed to evaluate and sort words by the intensity of the exceptional symmetry phenomenon [2]. For $w \in G_m = \{w_1, w_2, w_3, ..., w_L\}$,

$$R(w) = \left(\frac{\left(n_w - \frac{n_m}{L}\right)^2}{\frac{n_m}{L}} \right) / \left(\frac{(n_w - n_{w'})^2}{2(n_w + n_{w'})} \right), \tag{1}$$

with L the number of different words in G_m.

One disadvantage of the R measure is related to the unequal evaluation of dissimilarities between symmetric word pairs inside a ECG. If we have two pairs of words in the same ECG with identical dissimilarities between their occurrence frequencies, the R values could present distinct symmetry effects.

Consider for example $G_m = \{w_1, w_2, w_3, w'_1, w'_2, w'_3\}$, with $n_{w_1} = n_{w'_1} + 1 = 20$, $n_{w_2} = n_{w'_2} + 1 = 9$ and $n_{w_3} = n_{w'_3} + 1 = 1$. All symmetric word pairs present identical dissimilarities. The average frequency is $\frac{n_m}{L} = 9.5$ and the R values are

Table 2 Example to elucidate the differences between R and S measures.

words (w)	w_1	w'_1	w_2	w'_2	w_3	w'_3
n_w	20	19	9	8	0	1
R	905.2	741.0	0.9	8.1	19.0	15.2
$\ln(R)$	6.8	6.6	-0.1	2.1	2.9	2.7
S	2.5	2.5	2.5	2.5	2.5	2.5

presented in Tab. 2. The symmetric word pair with occurrences nearest to the average numbers of occurrences in the ECG is considered by the R measure to be less exceptional than the word pair whose number of occurrences is most distant. So, the R measure may be inadequate to sort the genomic words by exceptional symmetry. In order to avoid this disadvantage we introduce, in this study, a new measure S, the symmetric word pair effect,

$$S(w) = \ln \frac{\sqrt{\frac{\sum_{i=1}^{L}\sum_{j=1}^{L}(n_{w_i} - n_{w_j})^2}{L^2 - L}}}{|n_w - n_{w'}|}. \tag{2}$$

We can observe that in the numerator of the S measure we have the global deviation between G_m words. However, the numerator of the R measure is the deviation to the mean of the number of occurrence in G_m. Additionally, in the S measure the effect for both words of one symmetry word pair is the same, while the R measure can produce distinct values for each word of the symmetric pair. Table 2 presents the S values for the example discussed previously, and as expected, all words in this word group show equal exceptional symmetry effect.

We calculate the $S(w)$ value only for non SSW, because the exceptional symmetry of SSW is naturally infinitely high. When $n_w = n'_w$, we obtain $S(w) = \infty$. In our analysis the infinity is replaced, when necessary, by the double of the maximum effect obtained for the other words of the same length in the same species symbolizing a high exceptional symmetry value.

2.1 Control Experiments

In order to evaluate the exceptional symmetry in each word we generate random sequences under second parity rule validity assumption, i.e., using the same composition for complementary nucleotides. We generate the nucleotide sequences assuming nucleotide independence. In this scenario the expected probabilities of the words in each ECG are the same (see details in [1]). We denote these random sequences by *sym*.

2.2 Word Analysis Procedure

To identify a symmetric word pair as exceptional we compare the S values with the critical values obtained by the control experiments. To find words with very exceptional symmetry we use the third quartile has a cutoff threshold.

To compare genomes we use hierarchical clustering. We use a hierarchical bi-clustering procedure to compare both genomes and word S values, simultaneously. Hierarchical clustering was obtained using the UPGMA aggregation criterion with Euclidean distance.

3 Results and Discussion

We consider all genomic words with lengths k ($k \in \{1, ..., 12\}$) and a set of 31 genomes, and we obtain the symmetric word pair effect for each genomic word. As obvious result for $k = 1$, $S(w) = 1$ for all nucleotides.

Table 3 Percentage of words with exceptional symmetry effect ($S > 0$).

k (%)	2	3	4	5	6	7	8	9	10	11	12
HSap	100	100	100	100	100	100	100	100	100	100	100
MacM	100	100	100	100	100	100	100	100	100	100	100
PanT	100	100	100	100	100	100	100	100	100	100	100
MusM	100	100	100	100	100	100	100	100	100	100	100
RatN	100	100	100	100	100	100	100	100	100	100	100
DRer	100	100	100	100	99	99	98	98	99	100	100
Apis	100	100	100	100	100	100	100	51	11	79	99
CaeE	100	100	100	100	100	100	100	100	100	100	100
DroM	100	100	100	100	100	100	100	100	100	100	100
AraT	100	100	100	100	100	100	100	100	100	100	100
VitV	100	100	100	100	100	100	100	100	100	100	100
SacC	100	100	100	100	100	100	100	99	99	100	100
CanA	100	100	100	100	100	100	98	98	99	100	100
PlaF	100	100	100	100	100	100	100	99	100	100	100
HelP	100	100	100	100	100	100	100	99	99	100	100
StMG	100	100	100	100	100	100	99	97	98	100	100
StML	100	100	100	100	100	100	99	96	98	100	100
StPn	100	100	100	100	100	100	99	97	98	100	100
EscC	100	100	100	100	100	100	100	100	100	100	100
AerC	100	100	100	100	100	100	99	98	99	100	100
AerP	100	100	100	100	100	100	99	98	99	100	100
CalL	100	100	100	100	100	100	98	98	99	100	100
CanK	100	100	100	100	100	100	100	99	99	100	100
NanE	100	100	100	100	100	99	95	95	99	100	100
AbaS	100	97	99	98	91	82	80	91	99	100	100
AcaT	100	100	100	99	98	94	88	88	96	100	100
AchD	63	75	81	77	78	73	87	98	100	100	100
AcPL	63	69	78	78	73	74	84	95	99	100	100
AcPM	50	66	70	71	78	80	90	98	100	100	100
SouT	63	63	71	75	76	70	90	99	100	100	100
ZikV	63	78	83	83	81	79	76	97	100	100	100
sym	63	72	75	73	70	69	69	84	97	100	100

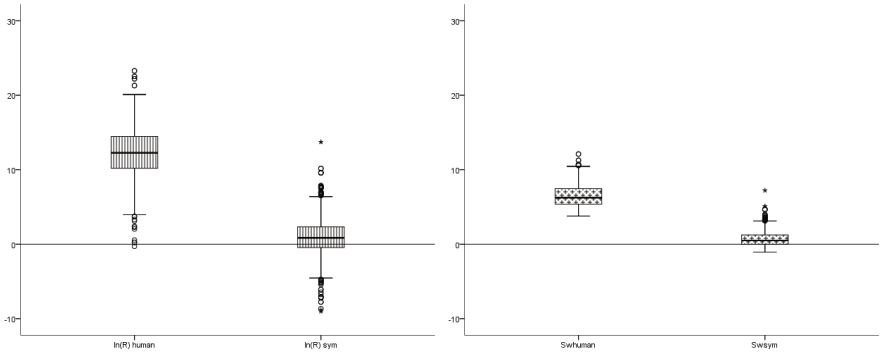


Fig. 1 Boxplots for S (left) and $\log R$ (right) values of human genome and random sequence example (*sym*) for word length 5.

Almost all words in eukaryote genomes show significant exceptional symmetry effect (comparing with the critical values obtained by simulation). Most of the genomic words have some degree of exceptional symmetry $S > 0$. Table 3 shows the percentage of words with $S > 0$ for each species and word length of this study. The viruses present a high percentage of words without exceptional symmetry. However, this result is expectable, since in a previous work, using a global measure of symmetry for all word of a fixed word length [1], it was concluded that some viruses do not have significant exceptional symmetry. Note that, with the increase of the word length some species reveal several symmetric word pairs where both elements have no occurrences. We considered these cases as having exceptional symmetry effect. This option artificially increases the percentage of words with exceptional symmetry in longer words for the species with shorter genomes.

To show the differences between real genomes and the random sequences generated under the second parity rule assumption, Table 3 includes the *sym* row corresponding to one control scenario (sequence with length equal to the length of human genome). Figure 1 shows the S (left) and $\ln R$ (right) values for $k = 5$ in the human genome and boxplots for the corresponding random realization *sym*. The boxplot for the human genome shows high symmetric word pair effects in both measures. The right outliers detected in the human S boxplot are: (GCGTA, TACGC), (ACCGG, CCGGT), (GCCAC, GTGGC), (GCCCA, TGGGC), (CGGGA, TCCCG). And the right outliers detected in the human boxplot of $\ln R(w)$ values are: $TACGC$, $GCGTA$, $TGGGC$, $GCCCA$, $GTGGC$, $GCCAC$. We note that, in this genome, the $R(w)$ outliers are a subset of the S outliers. The S and $\ln R$ measures differ the most in the detection of non symmetric words: S shows exceptional symmetry for every length-5 word, whereas $\ln R$ detects some word pairs with no exceptional symmetry. Due to the R disadvantages discussed in the Methods section we consider S measure results as more reliable.

Figure 2 shows the dendrogram obtained using the UPGMA aggregation criteria with Euclidean distance for $k = 5$. We observe three distinct groups: mammalian

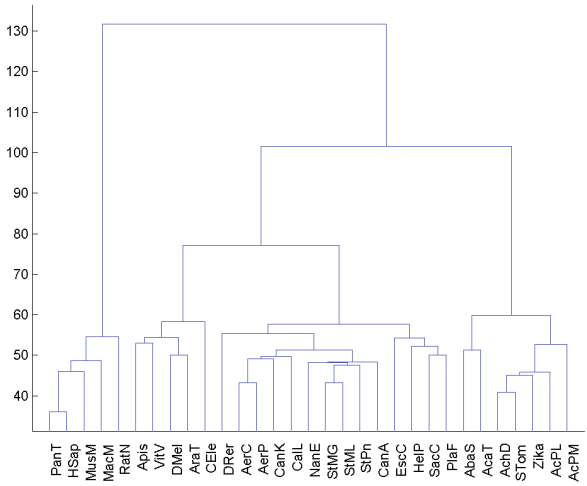


Fig. 2 Dendrogram of S values for all species under study word length 5.

species, viruses and the other species. The dendrograms obtained for other word lengths essentially maintain the same structure (the dendrogram for $k = 3$ is also included in Figure 3). Figure 3 shows the colormap with biclustering organization for $k = 3$. The cluster for the species highlights the viruses and a subgroup of animals. The symmetric word pair effect is stronger on the left side of the colormap and weaker on the right side. The word cluster highlights the group compound by two symmetric word pairs: (CCG, CGG), (GCG, CGC). We analysed the words with

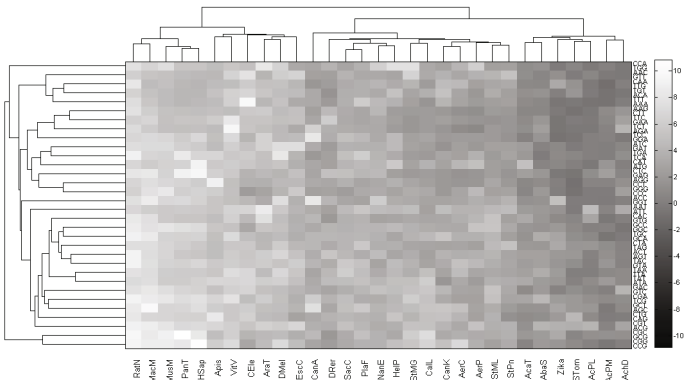


Fig. 3 Colormap with biclustering organization of S values for all species under study and word length 3.

most exceptional symmetry in all species under study, the words with exceptional symmetry effect above the third quartile. Above the third quartile we observe no common word for all the species under analysis. However, we observe some common words in animals group. The strongest symmetric word pair effect is observed in words composed by CpG dinucleotides.

4 Conclusions

We proposed a new measure to evaluate the exceptional symmetry effect. The exceptional symmetry values seem to contain information about the species evolution. The eukaryote group showed the highest exceptional symmetry in this study and the mammalian species has very high exceptional symmetry values distinct from all other species under study.

All cellular organisms under study present high percentages of words with exceptional symmetry effect. We conjecture that exceptional symmetry is an universal law of cellular organisms, but the most exceptional symmetric words are species specific.

We reinforce that some viruses show a behavior opposite to exceptional symmetry in almost all words under study ($S < 0$).

Acknowledgment This work was supported by Portuguese funds through the iBiMED - Institute of Biomedicine, IEETA - Institute of Electronics and Telematics Engineering of Aveiro and the Portuguese Foundation for Science and Technology ("FCT–Fundação para a Ciência e a Tecnologia"), within projects: UID/BIM/04501/2013 and PEst-OE/EEI/UI0127/2014.

References

1. Afreixo, V., Rodrigues, J.M.O.S., Bastos, C.A.C.: Exceptional single strand DNA word symmetry: analysis of evolutionary potentialities. Journal of Integrative Bioinformatics **11**(3), 250 (2014)
2. Afreixo, V., Rodrigues, J.M.O.S., Bastos, C.A.C.: Analysis of single-strand exceptional word symmetry in the human genome: new measures. Biostatistics **16**(2), 209–221 (2015)
3. Albrecht-Buehler, G.: Inversions and inverted transpositions as the basis for an almost universal "format" of genome sequences. Genomics **90**, 297–305 (2007)
4. Baisnée, P.-F., Hampson, S., Baldi, P.: Why are complementary DNA strands symmetric? Bioinformatics **18**(8), 1021–1033 (2002)
5. Chargaff, E.: Chemical specificity of nucleic acids and mechanism of their enzymatic degradation. Experientia **6**(6), 201–209 (1950)
6. Forsdyke, D.R.: Evolutionary Bioinformatics. Springer, New York (2011)
7. Forsdyke, D.R., Bell, S.J.: Purine loading, stem-loops and Chargaff's second parity rule: a discussion of the application of elementary principles to early chemical observations. Applied Bioinformatics **3**(1), 3–8 (2004)
8. Karkas, J.D., Rudner, R., Chargaff, E.: Separation of B. subtilis DNA into complementary strands. II. template functions and composition as determined by transcription with RNA polymerase. Proceedings of the National Academy of Sciences of the United States of America **60**(3), 915–920 (1968)

9. Powdel, B.R., Satapathy, S.S., Kumar, A., Jha, P.K., Buragohain, A.K., Borah, M., Ray, S.K.: A study in entire chromosomes of violations of the intra-strand parity of complementary nucleotides (chargaff's second parity rule). DNA Research **16**, 325–343 (2009)
10. Rudner, R., Karkas, J.D., Chargaff, E.: Separation of B. subtilis DNA into complementary strands, I. biological properties. Proceedings of the National Academy of Sciences of the United States of America **60**(2), 630–635 (1968)
11. Rudner, R., Karkas, J.D., Chargaff, E.: Separation of B. subtilis DNA into complementary strands. III. direct analysis. Proceedings of the National Academy of Sciences of the United States of America **60**(3), 921–922 (1968)

A Computation Tool for the Estimation of Biomass Composition from Genomic and Transcriptomic Information

Sophia Santos and Isabel Rocha

Abstract Given the huge potential impact of the growing number of complete genome-scale metabolic network reconstructions of organisms, it is necessary to use bioinformatics tools to simplify and accelerate the course of knowledge in this field. One essential component of genome-scale metabolic model is biomass equation, whose maximization is one of the most common objective functions used in Flux Balance Analysis formulations. Some components of biomass, such as amino acids and nucleotides, can be estimated from genome information. In this work it is proposed a Java tool that estimates biomass composition in amino acids and nucleotides, from genome and transcriptomic information, using as input files sequences in FASTA format and files with transcriptomic data in csv format. This application allows to obtain the results rapidly and is also a user-friendly tool since facilitates its use by users with any or little background in informatics.

Keywords Genome-scale metabolic models · Biomass equation · Genome information · Transcriptomic information · Amino acids · Deoxynucleotides and nucleotides

1 Introduction

Genome-scale metabolic models are a valuable tool for the study of metabolic systems [14] and are becoming available for an increasing number of organisms [13]. These network reconstructions are used to compute a variety of phenotypic states [5] in order to implement metabolic engineering strategies, or identify drug-targets, among other applications [9]. Flux balance analysis (FBA) is a mathematical widely used approach for genome-scale simulation of metabolic fluxes. FBA uses linear optimization to determine one steady-state reaction flux distribution in a metabolic

S. Santos · I. Rocha(✉)
Center of Biological Engineering, University of Minho, Braga, Portugal
e-mail: irocha@deb.uminho.pt

M.S. Mohamad et al. (eds.), *10th International Conference on PACBB*,
Advances in Intelligent Systems and Computing 477,
DOI: 10.1007/978-3-319-40126-3_17

network by maximizing an objective function. The most common objective function involves the maximization of biomass formation, which has proven to be consistent with experimental observations in several conditions [4]. The formulation of the biomass composition to be used as objective function can be performed at different levels of detail: basic level (defining the macromolecular content on the cell, i.e., protein, RNA, DNA, lipids), intermediate level (calculating the necessary biosynthetic energy) and advanced level (detailing the necessary vitamins, elements, and cofactors) [5]. For n biomass constituents, the biomass formation equation can be formulated as:

$$\sum_{i=1}^{n} c_i X_i \rightarrow Biomass \tag{1}$$

where c_i is the coefficient of each component, X_i, considered in the biomass. The units of all the coefficients are defined in mmol per gram of dry weight (mmol/gDW) and the biomass formation units are defined per hour (h^{-1}). If a biomass component is not accounted for in the biomass objective function, the corresponding synthesis reactions may not be required for growth, as well as the associated genes. Thus, the composition of biomass plays an important role for example for in silico predictions of essencial genes [14]. In order to achieve good predictions, a detailed biomass composition of an organism needs to be experimentally determined for cells growing in log phase using available methods. However, often experimental methods are laborious and time consuming or the modelled oraganism is difficult to grow in the lab. In many cases, when no experimental data are available, biomass composition of related organisms is included in the model. Also, some components such as amino acids, nucleotides (NTPs) and deoxynucleotides (dNTPs) can be estimated from genome information, as described in 2010 by Thiele and Palsson in their detailed protocol to create a genome-scale metabolic network reconstruction. Some studies indicate that this approach is more reliable than performing aproximations to closely related organisms [12]. However, when estimating amino acids compositions directly from the genome, it is assumed that all proteins are being expressed at all times, in the same proportions, a fact that is known to be false. Indeed, some authors already have used genome information allied with transcriptomic data in other to estimate more accurately biomass amino acid composition [11]. In this work a java tool was developed, which returns the estimated biomass composition in amino acids, NTPs and dNTPs, from files with selected sequences and transcriptomic data.

2 *In Silico* Biomass Determination

2.1 Genome Information

As Thiele and Palsson (2010) have indicated, the estimation of the composition in amino acids, NTPs and dNTPs from genome information can be performed by

calculating the percentage of each monomer and converting it into mmol/gDW. Genome information is easily found in databases and can be extracted in various formats, like FASTA and GENBANK format. In order to determine the NTPs composition of the cell, the protocol described by Thiele and Palsson uses the codon usage accessed for the amino acid content. Since RNA incorporates uracil (U) instead of thymine (T), the codon usage needs to be read with every T replaced by a U. However, in this report the authors do not distinguish between the different types of RNA and, as a result, perform their calculations for messenger RNA (mRNA) only. However, in a prokaryotic cell 95% of total RNA is transfer RNA (tRNA) and ribosomal RNA (rRNA) [10], and therefore only less that 5% of all RNA is messenger RNA (mRNA). Therefore, some changes were made to the protocol described by Thiele and Palsson regarding NTPs estimation. Genome information for mRNA, rRNA and tRNA is used in the new protocol and the NTPs are determined taking into account the percentage of each molecule in the total RNA. These percentages differ also among organisms: gram positive bacteria have on average 5% mRNA, 20% tRNA and 75% rRNA and gram negative bacteria and yeast have 5% mRNA, 15% tRNA and 80% rRNA [8, 15].

2.2 *Transcriptomic Information*

To determine the amino acid composition in biomass gene expression data can also be used together with genome sequencing information, as long as these data are available for awide variety of relevant conditions. Gene expression data should be available as total abundance of expression of each gene/protein, witch needs to be normalized to a ratio (referred in equation (2) as $Abundance^p$ or abundance of protein p). The composition of each protein in amino acid i (AA^p) is taken from the genome information (being $AA^{p,g}$ the amino acid composition obtained from the genome) and is corrected by the expression factor as shown in equation (2).

$$AA_i^p = AA_i^{p,g}(ratio) \times Abundance^p(ratio) \tag{2}$$

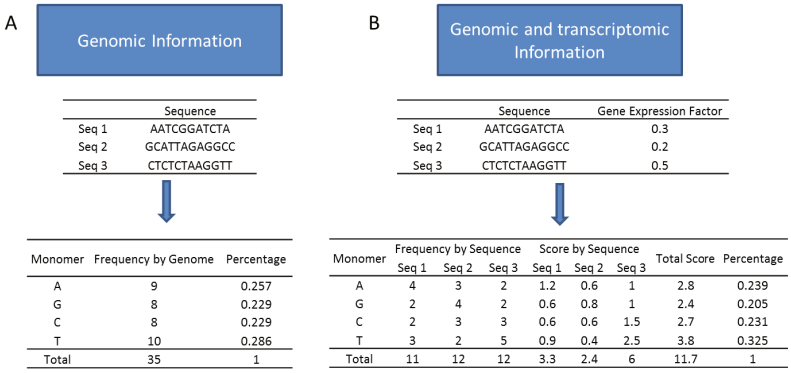
The total biomass content in each amino acid i is determined by the sum of values of each amino acid for all proteins (P) divided by the sum of all amino acids (N) for all proteins:

$$AA_i^T(\%) = \frac{\sum_p^P AA_i^p}{\sum_{p,i}^{P,N} AA_i^p} \tag{3}$$

The values obtained are expressed in molar percentage, and have to be converted in to mmol/gDW to be included in the biomass equation.

3 Computational Tool for the Estimation of Biomass Composition from Genome Data

The application developed is fully implemented in the Java language, and using BioJava packages. The main capability of the application is the estimation of biomass composition in amino acids and nucleotides from the genome and transcriptomic information, using as input files sequences in FASTA format and expression data in csv format. The application determines the percentage of each amino acid and nucleotide in the cell, as exemplified in Fig. 1, and also the same value in mmol/gDW, to directly add to the biomass equation.



	Sequence
Seq 1	AATCGGATCTA
Seq 2	GCATTAGAGGCC
Seq 3	CTCTCTAAGGTT

	Sequence	Gene Expression Factor
Seq 1	AATCGGATCTA	0.3
Seq 2	GCATTAGAGGCC	0.2
Seq 3	CTCTCTAAGGTT	0.5

Monomer	Frequency by Genome	Percentage
A	9	0.257
G	8	0.229
C	8	0.229
T	10	0.286
Total	35	1

Monomer	Frequency by Sequence			Score by Sequence			Total Score	Percentage
	Seq 1	Seq 2	Seq 3	Seq 1	Seq 2	Seq 3		
A	4	3	2	1.2	0.6	1	2.8	0.239
G	2	4	2	0.6	0.8	1	2.4	0.205
C	2	3	3	0.6	0.6	1.5	2.7	0.231
T	3	2	5	0.9	0.4	2.5	3.8	0.325
Total	11	12	12	3.3	2.4	6	11.7	1

Fig. 1 Process to determine percentage of each monomer (A) from genome information and (B) from genome and transcriptomic information

This application allows obtaining rapidly this kind of information and it is also a user-friendly tool, facilitating its use by operators with any or little background in informatics. The application is separated by three tabs, one to estimate the Protein composition in amino acids (Fig. 2 A), one to estimate the DNA composition in dNTPs (Fig. 2 B), and another to estimate the RNA composition in NTPs (Fig. 2 C). To use the application, it is indispensable to input files with sequences of Protein, DNA and RNA, exclusive in the FASTA format. If transcriptomic data are available for the organism in study they can be added in the csv format, with two columns separated by semicolon: (the first with gene identifiers and the second with the expression factor in percentage). In this case the FASTA file with protein sequences should have the same identifier at the beginning of the sequence header. To obtain the results it is just necessary to click in the **Determine** button.

The application requires some obligatory inputs: percentage of each type of RNA (mRNA, rRNA and tRNA), that is specific for each organism, and also the value of the cellular content in each macromolecule (Protein, DNA and RNA) in percentage, in order to calculate the corresponding biomass composition in mmol/gDW.

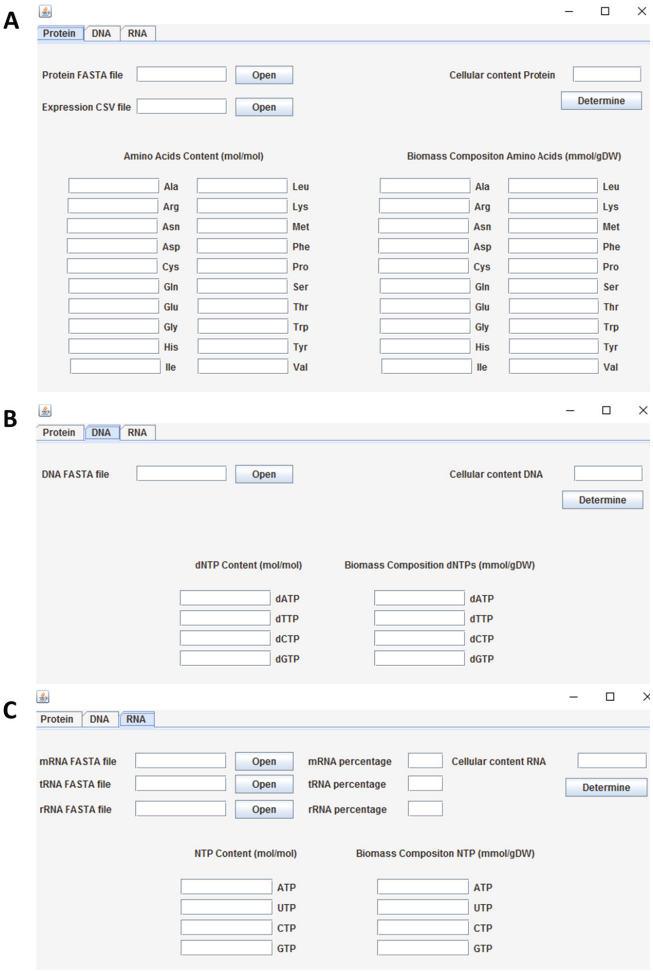


Fig. 2 Screen-shots from the Java application developed. (A) Tab to estimate the Protein composition in amino acids, (B) tab to estimate the DNA composition in dNTPs, and (C) tab to estimate the RNA composition in NTPs and the corresponding biomass composition for the three components.

4 Case-Study

To illustrate some of the main features of the java tool, the genome sequence [1] and transcriptomic information [3] of the common laboratory strain *Escherichia coli* K-12 MG1655 were used. The results obtained were compared with experimental data. All FASTA files with genome information were retrieved from the NCBI database [2].

4.1 Amino Acid Composition

To determine amino acid composition, the Protein FASTA file had to be adapted to have in each sequence header the same identifier of the transcriptomic data file for each protein/gene (in this case locus tags identifiers) as seen in Fig. 3.

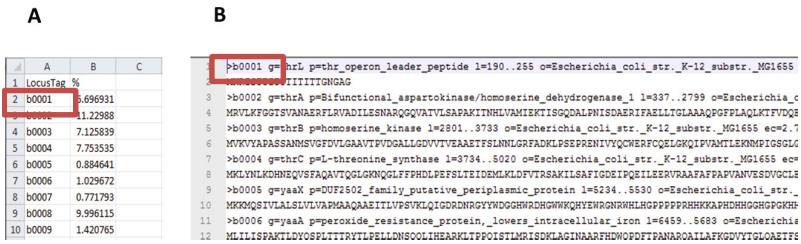


Fig. 3 Screen-shots from the files used to estimate amino acid composition. (A) Csv file with gene/protein identifier and respective expression factor and (B) FASTA file with all protein sequences.

The results obtained for amino acid composition are summarized in Table 1 and are also compared to experimental data [8]. The results obtained for the amino acid composition are fairly close to the experimental data. As expected, the amino acid composition obtained using genome and transcriptomic data is closer to experimental data than the amino acid composition using only genome information. There are some differences between experimental data and estimated data for some amino acids, such as aspartate, asparagine and leucine. These differences may probably result by the fact that the experimental data can be affected by experimental errors and also the fact that some amino acids are more sensitive to the experimental methods than others. Some amino acids are very sensitive to hydrolysis, particularly cysteine, tryptophan and methionine [6], while asparagine is transformed in aspartic acid and glutamine is transformed in glutamic acid. Other amino acids are more sensitive to the derivatization step, producing more than one derivative, such as glycine and lysine, or by losing detectable response, such as leucine [7]. However, in all cases the estimated data is very close to experimental data.

4.2 Nucleotide and Deoxynucleotide Compositions

The results obtained for NTPs and dNTPs compositions are summarized in Table 2 and are also compared with experimental data.

The values for the estimation of biomass composition in NTPs are not as close to experimental data as expected. These differences are probably the result of

Table 1 Comparison of amino acid composition obtained using the Java tool with only genome information and using genome and transcriptomic information with experimental data from the literature. The data are presented in molar percentage.

Amino Acid	Experimental [8]	Genome	Genome & Transcriptomic
Ala	9.60	9.55	9.79
Arg	5.53	5.53	5.59
Asn	4.51	3.89	3.84
Asp	4.51	5.13	5.44
Cys	1.71	1.16	1.03
Gln	4.92	4.45	4.31
Glu	4.92	5.78	6.35
Gly	11.45	7.37	7.63
His	1.77	2.27	2.20
Ile	5.43	6.01	5.90
Leu	8.42	10.71	10.06
Lys	6.42	4.39	4.93
Met	2.87	2.83	2.80
Phe	3.46	3.90	3.69
Pro	4.13	4.44	4.32
Ser	4.03	5.77	5.38
Thr	4.74	5.37	5.29
Trp	1.06	1.53	1.29
Tyr	2.58	2.83	2.73
Val	7.91	7.09	7.41
Correlation to Experimental Data (R^2)		0.759	0.806

Table 2 Nucleotide and deoxynucleotide compositions estimated from genome information. The data is presented in molar percentage.

Nucleotide or Deoxynucleotide	Experimental [8]	Genome
ATP	20.00	23.50
CTP	32.22	25.69
GTP	21.59	28.97
UTP	26.19	21.84
dATP	24.60	24.62
dCTP	25.40	25.42
dGTP	25.40	25.37
dTTP	24.60	24.59

uncertainties in the percentage of each type of RNA molecule in the total RNA composition. The values estimated for the biomass composition in dNTPs are very similar to the reference data.

5 Conclusions

The Java application created is a Java tool that provides an estimation of biomass composition in nucleotides and amino acids, with input files containing genome sequences from DNA, RNA and protein, in the FASTA format. When available expression data can also be used, provided a csv file containing percentages of each gene/protein. The results are fairly close to experimental data showing that the estimation of amino acid and nucleotide compositions from genome information and from transcriptomic data is a good alternative when no experimental data is available.

Acknowledgments The authors thank the project DeYeastLibrary - Designer yeast strain library optimized for metabolic engineering applications, Ref. ERA-IB-2/0003/2013, funded by national funds through FCT/MCTES.

References

1. Blattner, F.R., Plunkett, G., Bloch, C.A., Perna, N.T., Burland, V., Riley, M., Collado-Vides, J., Glasner, J.D., Rode, C.K., Mayhew, G.F., et al.: The complete genome sequence of Escherichia coli K-12. Science **277**(5331), 1453–1462 (1997)
2. Coordinators, N.R.: Database resources of the national center for biotechnology information. Nucleic Acids Research **41**(D1), D8–D20 (2013)
3. Covert, M.W., Knight, E.M., Reed, J.L., Herrgard, M.J., Palsson, B.O.: Integrating high-throughput and computational data elucidates bacterial networks. Nature **429**(6987), 92–96 (2004)
4. Edwards, J.S., Ibarra, R.U., Palsson, B.O.: In silico predictions of Escherichia coli metabolic capabilities are consistent with experimental data. Nature Biotechnology **19**(2), 125–130 (2001)
5. Feist, A.M., Palsson, B.O.: The biomass objective function. Current Opinion in Microbiology **13**(3), 344–349 (2010)
6. Fountoulakis, M., Lahm, H.-W.: Hydrolysis and amino acid composition analysis of proteins. Journal of Chromatography A **826**(2), 109–134 (1998)
7. Molnar-Perl, I.: Derivatization and chromatographic behavior of the o-phthaldialdehyde amino acid derivatives obtained with various SH-group-containing additives. Journal of Chromatography A **913**(12), 283–302 (2001)
8. Neidhardt, F.C., Curtiss, R.: Escherichia coli and Salmonella : cellular and molecular biology, vol. 2, 2nd edn. ASM Press, Washington, D.C. (1996)
9. Raman, K., Chandra, N.: Flux balance analysis of biological systems: applications and challenges. Briefings in Bioinformatics **10**(4), 435–449 (2009)
10. Rosenow, C., Saxena, R.M., Durst, M., Gingeras, T.R.: Prokaryotic RNA preparation methods useful for high density array analysis: comparison of two approaches. Nucleic Acids Research **29**(22), e112–e112 (2001)
11. Saha, R., Verseput, A.T., Berla, B.M., Mueller, T.J., Pakrasi, H.B., Maranas, C.D.: Reconstruction and comparison of the metabolic potential of cyanobacteria Cyanothece sp. ATCC 51142 and Synechocystis sp. PCC 6803. PloS one **7**(10), e48285 (2012)
12. Santos, S.T.: Development of computational methods for the determination of biomass composition and evaluation of its impact in genome-scale models predictions. Master's thesis, Universidade do Minho (2013)

13. Simeonidis, E., Price, N.D.: Genome-scale modeling for metabolic engineering. Journal of Industrial Microbiology & Biotechnology **42**(3), 327–338 (2015)
14. Thiele, I., Palsson, B.Ø.: A protocol for generating a high-quality genome-scale metabolic reconstruction. Nature Protocols **5**(1), 93–121 (2010)
15. Warner, J.R.: The economics of ribosome biosynthesis in yeast. Trends in Biochemical Sciences **24**(11), 437–440 (1999)

Part IV
Systems Biology

Role of Nerve Growth Factor Signaling in Cancer Cell Proliferation and Survival Using a Reachability Analysis Approach

Gustavo Santos-García, Carolyn Talcott, Adrián Riesco, Beatriz Santos-Buitrago and Javier De Las Rivas

Abstract Systems biology attempts to understand biological systems by their structure, dynamics, and control methods. Nerve growth factor (NGF) is a neuropeptide involved in cellular signaling that binds specific cell surface receptors in order to induce cellular proliferation and survival in different cell types or cell contexts. In this paper we perform a reachability analysis and we compute common elements in all possible solutions in our cases of interest with the help of Pathway Logic, which constitutes a rewriting logic formalism that provides a knowledge base and development environment to carry out model checking, searches, and executions of signaling systems. In conclusion, we provide a symbolic system that explores complex and dynamic cellular signaling processes that induce cellular proliferation and cellular survival.

G. Santos-García(✉)
University of Salamanca, Salamanca, Spain
e-mail: santos@usal.es

C. Talcott
Computer Science Laboratory, SRI International, Menlo Park, USA
e-mail: clt@csl.sri.com

A. Riesco
Universidad Complutense de Madrid, Madrid, Spain
e-mail: ariesco@ucm.es

B. Santos-Buitrago
Seoul National University, Seoul, South Korea
e-mail: bsantosb@snu.ac.kr

J. De Las Rivas
Cancer Research Center (CSIC/USAL) and IBSAL, Salamanca, Spain
e-mail: jrivas@usal.es

Pathway Logic development has been funded in part by NIH BISTI R21/R33 grant (GM068146-01), NIH/NCI P50 grant (CA112970-01), and NSF grant IIS-0513857. This work was partially supported by NSF grant CNS-1318848. Research was supported by Spanish projects Strongsoft TIN2012-39391-C04-04, TRACES TIN2015-67522-C3-3-R, and PI12/00624 (MINECO, Instituto de Salud Carlos III) and Comunidad de Madrid project N-Greens Software-CM (S2013/ICE-2731).

M.S. Mohamad et al. (eds.), *10th International Conference on PACBB*,
Advances in Intelligent Systems and Computing 477,
DOI: 10.1007/978-3-319-40126-3_18

Keywords Signal transduction · Symbolic systems biology · Nerve growth factor · Pathway logic · Rewriting logic · Maude

1 Modeling Signaling Networks

Systems biology is an emergent field that facilitates understanding biological systems by describing their structure, dynamics, and control methods. Investigation of mammalian signaling processes, the molecular pathways by which cells detect, convert, and internally transmit information from their environment to intracellular targets such as the genome, would greatly benefit from the availability of predictive models [3, 14]. Various models for computational analysis of cellular signaling networks have been proposed to simulate responses to specific stimuli [10, 11]. However, in many cases complex cell signaling pathways have to be treated with other more qualitative modeling approaches, like logic modeling.

Symbolic models allow us to represent partial information and to model and analyze systems at multiple levels of detail, depending on the information available and the questions to be studied. Such models are based on formalisms that provide a language for representing system states and mechanisms of change such as reactions, and analysis tools based on computational or logical inference. Symbolic models can be used for simulation of system behavior.

Nerve growth factor (NGF) cellular signaling is involved in the regulation of development, maintenance, growth, proliferation, survival, and death of certain neurons [8]. This signaling is initiated by binding the ligand to two membrane-bound receptors (the tropomyosine receptor kinase A, `TrkA`, and the low-affinity NGF receptor, `NGFR`) so as to trigger a cellular signaling path (Fig. 1).

In this paper, Section 1 gives an overview to logical modeling of biological systems with rewriting logic and Pathway Logic. In Sections 2 and 3, we show the implementation of various rules and advanced logical inferences in the NGF signaling pathway. Conclusions are drawn in Section 4.

Pathway Logic Models of Signaling Networks in Cells. Rule-based modeling allows us to intuitively specify biological interactions while abstracting from the underlying combinatorial complexity. Since Ordinary Differential Equations (ODE) based systems [4] are considered in general the standard model for analyzing systems biology, it is worth briefly discussing the differences between these approaches. From the reactions point of view, ODE-based frameworks focus on diffusion-like reactions, while rule-based frameworks rely on the concept of reactive molecular collisions. On the other hand, ODE-based approaches model the average behavior of a system where the time evolution is a continuos process, while rule-based approaches model individual runs that reach different states non-deterministically chosen. Finally, ODE-based analyses are able to deal with huge numbers of molecules per species, but their complexity becomes too complex when the number of molecules is huge. Contrarily, rule-based approaches deal easily with many molecular species, but cannot deal

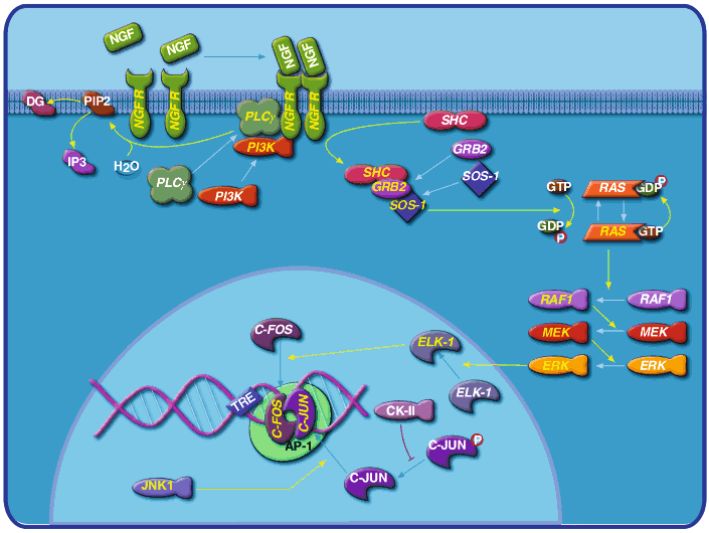


Fig. 1 Signaling of nerve growth factor pathway inside the cell (reprinted from [9]).

with very large number of molecules per species. In this work we are interested in using rule-based systems, since we want to analyze how particular dishes evolve and the particular states reachable from these dishes.[1]

Pathway Logic [12] is an approach to the modeling and analysis of molecular and cellular processes based on rewriting logic. Pathway Logic models of biological processes are developed using the Maude language. A Pathway Logic knowledge base includes data types representing cellular components such as proteins, small molecules, or complexes; compartments/locations; post-translational modifications and other dynamic events occurring in cellular reactions. The naturalness of rewriting logic for modeling and experimenting with mathematical and biological problems has been illustrated in a number of works [6, 7]. The basic idea is that we can model a cell as a concurrent system whose concurrent transitions are precisely its biochemical reactions [13]. In this way we can develop symbolic models of biological systems which can be analyzed like any other rewrite theory.

Maude [1, 2] is a high performance language and system supporting both equational and rewriting logic computation. Maude programs achieve a good agreement between mathematical and operational semantics. There are three different uses of Maude modules: (1) as programs that solve some problems; (2) as formal executable specifications that provide a rigorous mathematical model of an algorithm, a system, a language, or a formalism; and (3) as models that can be formally analyzed and verified with respect to different properties expressing various formal requirements.

[1] We use the term *dish* to refer to an initial state in analogy to an experimental setup in a Petri dish.

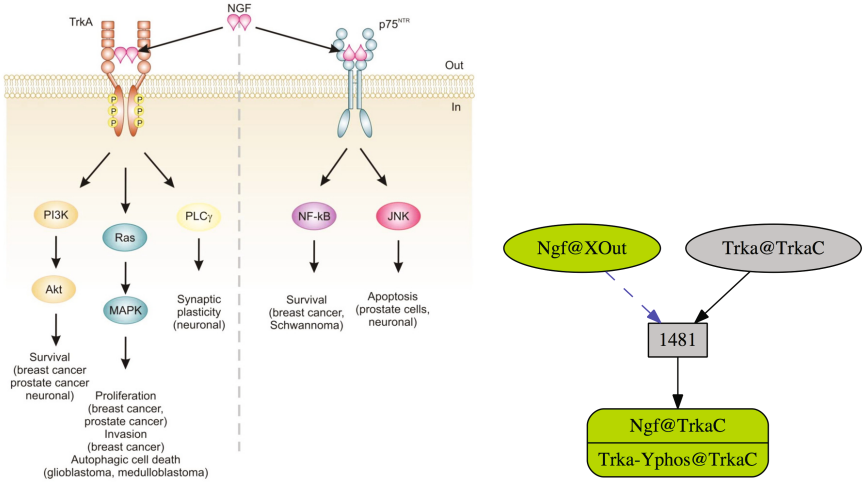


Fig. 2 (a) NGF binding to TrkA receptor mediates proliferation, differentiation and survival via activation of PI3K/Akt, Ras/MAPK and PLCγ pathways (cf. [8, Figure 1]). (b) Rule `[1481.Trka.irt.Ngf]` using Pathway Logic Assistant.

The Pathway Logic system, its documentation, a collection of examples, and related papers are available on http://pl.csl.sri.com. Models of cellular response to many different stimuli can also be found on our website.

2 Case Study: Modeling and Dynamics of NGF Signaling Pathway

2.1 Modeling: Dishes and Rewrite Rules

In this section we define some dishes and rules of the STM7 Pathway Logic knowledge base. A formal knowledge base contains information about the changes that occur in the proteins inside a cell in response to exposure to receptor ligands, chemicals, or various stresses. In our case study we will focus on models of response to nerve growth factor (NGF) stimulation. NGF cellular signaling is involved in the regulation of development, maintenance, growth, proliferation, survival, and death of certain nerve cells (neurons). The NGF signaling pathway includes reactions and circuits and, in fact, can induce cellular proliferation activating proteins ERK and AKT inside the cells (Fig. 2a).

An initial state or *dish* (called `NgfDish`) with several locations and elements is defined: the membrane (location tag `CLm`) is empty; the inside of the membrane (location tag `CLi`) contains proteins `Hras`, `Rap1a`, and `Rit1` binding to `GDP`;

the cytoplasm (location tag CLc) contains enzyme Pi3k and proteins Akt1, Aps, Araf, Arms, etc. (see the full code below); and the nucleus (location tag NUc) contains some genes (Egr1-gene and Fos-gene) and proteins (Elk1, Foxo1, and Foxo3). Moreover, there are three other locations: the outside (location tag XOut) which contains the nerve growth factor (Ngf), the NgfRC location which contains the common p75 neurotrophin nerve growth factor receptor (NgfR or $p75^{NTR}$), and the TrkaC location which contains the tropomyosin receptor kinase A (Trka):

```
eq NgfDish = PD({XOut | Ngf} {NgfRC | NgfR} {TrkaC | Trka}
  {CLm | empty}   {CLi | [Hras - GDP] [Rap1a - GDP] [Rit1 - GDP]}
  {CLc | Pi3k Akt1 Aps Araf Arms Braf Crk CrkI CrkL Erk5 Erks Frs2 Gab1
       Gab2 Ikba Ikk1 Ikk2 Irak1 Jnks Matk Mek1 Mek2 Myd88 P38s Pkcd
       Plcg1 Pxn Raf1 RapGef1 Rela Ripk2 Sh2b1 Shc1 Sqstm1 Traf6 Znf274}
  {NUc | Egr1-gene Fos-gene Elk1 Foxo1 Foxo3}) .
```

Rewrite rules describe the behavior of proteins and other components depending on modification states and biological contexts. Each rule represents a step in a biological process such as metabolic reactions or intra/inter cellular signaling reactions.

Rewrite Rule 1481. Pathway Logic contains a set of rules, derived from curated experimental findings, that provide a logical explanation of how a signal propagates in response to an NGF stimulus. Here we describe rule 1481, directly sourced from the literature. Hartman *et al.* [5] determine that PC12 cells express two distinct nerve growth factor receptors (NGFRs), p75NGFR and trkA (p140trk).

Our rewrite rule 1481 establishes: *In the presence of nerve growth factor Ngf in the outside of the cell (XOut), the protein Trka is phosphorylated on tyrosine ([Trka - Yphos]) and binds to NGF (Ngf).* In Maude syntax, this signaling process is described by the following rewrite rule:

```
rl [1481.Trka.irt.Ngf]:
   {XOut | xout Ngf}  {TrkaC | trkac Trka}
=> {XOut | xout Ngf}  {TrkaC | trkac ([Trka - Yphos] : Ngf)} .
```

Figure 2b shows this rule using the Pathway Logic Assistant. Ovals represent biomolecules participating in reactions (e.g., proteins, genes, etc.). Rectangles represent reaction rules with a label which represents its abbreviated identifier in the knowledge base. Solid arrows from an occurrence oval to a rule indicate that the occurrence is a reactant. Dashed arrows indicate that the occurrence is a modifier/enzyme/control, i.e., it is necessary for the reaction to take place but is not changed by the reaction. Solid arrows from a rule to an occurrence oval indicate that the occurrence is a product.

2.2 *Dynamics: Logical Inferences*

Thanks to the Knowledge Base provided by Pathway Logic, our analysis begins with the initial dish state `NgfDish` defined in Section 2.1. It is a well-known fact that cell proliferation and survival are connected to activation of `Akts` or `Erks`. We want to find out if there is a pathway from `NgfDish` leading to activation of `Akts` or `Erks`. In this case one can use the `search` command with a suitable search pattern and parameters (`[n]`: the first *n* solutions; `=>+`: at least one step). The target state is defined by the operator `PD`, whose argument is a "soup" of locations with their respective contents. A soup is a multiset that can include several elements regardless of their order.

The contents of each location (e.g., `XOut`) are things and/or variables (e.g., `thXOut:Things`) that contain elements according to the matching criteria of our search. In the cytoplasm, a protein `prot:BProtein` must be activated and can also have a set of other modifiers `mod:ModSet`. The search condition imposes that the variable `prot:BProtein` has membership in the sort `ErkS` or `AktS`.

```
Maude> search [1000] in QQ : NgfDish =>+ PD( loc:Locations
  {TrkaC | th:Things (Ngf : [Trka - Yphos])} {NgfRC | Ngf : NgfR}
  {XOut | thXOut:Things}  {CLm | thCLm:Things}
  {CLi | thCLi:Things}    {NUc | thNUc:Things}
  {CLc | thCLc:Things [prot:BProtein - mod:ModSet act]})
  such that (prot:BProtein :: ErkS  or  prot:BProtein :: AktS ) = true .
```

The solutions to this query given by Maude show the matching in the previous search pattern. While the terms fixed by the search pattern are not shown (e.g. `Ngf : NgfR`), the variables are presented with their corresponding values. For example, the 999th solution has the following values:

```
Solution 999 (state 48024)
loc:Locations --> empty
thXOut:Things --> Ngf
thCLm:Things --> empty
thCLi:Things --> [Hras - GDP] [Rap1a - GDP] [Rit1 - GDP]
thCLc:Things --> Aps Arms Crk CrkI CrkL Erks Erk5 Frs2 Gab1 Gab2 Ikba Ikk1
     Ikk2 Irak1 Jnks Matk Mek1 Mek2 Myd88 P38s Pi3k Pkcd Plcg1 Pxn Raf1
     RapGef1 Rela Ripk2 Shc1 Sqstm1 Traf6 [Araf - act] [Braf - act] Znf274
thNUc:Things --> Egr1-gene Fos-gene Elk1 Foxo1 Foxo3
th:Things --> [Sh2b1 - Yphos]
prot:BProtein --> Akt1
mod:ModSet --> none
```

In this solution, we observe that the variable `prot:BProtein` matches with protein `Akt1` without any set of modifications. We find out two proteins `Araf` and `Braf` in an activated form in the cytoplasm. We show evidence of a ligand/receptor effect: nerve growth factor (NGF) binds a specific cell surface receptor (NGFR). Then we ask Maude for the rule labels which have been applied to reach the final state according to the solution. One of these rules is the rewrite rule `1481.Trka.irt.Ngf` described above.

```
Maude> show path labels 48024 .
  1481.Trka.irt.Ngf    1482.NgfR.irt.Ngf    1484.Akt1.irt.NgfR.Ngf
  1488.Araf.irt.Ngf    1489.Braf.irt.Ngf    1517.Sh2b1.irt.Ngf
```

In this way, Maude allows us to explore the complete search space following a breadth-first strategy until all solutions are found.

3 Advanced Logical Inferences: Computing Similarities

A key distinguishing feature of Maude language is its systematic and efficient use of reflection (i.e. Maude's capability of handling and reasoning about terms that represent specifications described in Maude itself) through its predefined `META-LEVEL` module [2, Chapter 14]. Using the metalevel we can implement functions that manipulate and execute the modules given as argument, as well as the results obtained from these executions.

In this way, we have devised a method to compute all the solutions for a pathway given, the initial term `NgfDish`, a pattern and a condition for the reached terms, and a bound in the number of steps. Once all these terms have been computed it compares them in order to find the common structure underlying the solutions. That is, given the solutions `f(g(0), c, a)`, `f(b, c, g(c))`, and `f(b, c, g(c))`, it returns a skeleton `f(g(_), c)` that reveals the structure shared by all the solutions up to this depth.

Using this function we first tried to compute the similarities shared by all the solutions obtained for the search in the previous section. However, we realized that the state space was too big for this kind of abstraction, since too many reachable solutions are available and the relation between them is lost. Hence, we decided to start with low bounds for the depth of the search and progressively increase it to see how it evolves.

In particular, we have computed the common structure for the 536 solutions found when applying at most 5 rules and the 5718 solutions found for at most 6 rules. For the first search we obtain the following result:

```
Maude> red computeSimilarities(module, init, pattern, cond, '+, 5) .
PD( {XOut | Ngf} {NgfRC | Ngf : NgfR} {TrkaC | _} {CLm | empty}
{CLc | Ikk1 Ikk2 Irak1 Jnks Matk Myd88 P38s Pkcd Ripk2 Sqstm1 Traf6
     Znf274} {CLi | [Hras - _] [Rap1a - _] [Rit1 - _]}  {NUc | Elk1})
```

where the constants `module`, `init`, `pattern`, and `cond` stand for the module, initial term, pattern, and condition used in the previous search, respectively. Similarly, we have the following search for depth 6:

```
Maude> red computeSimilarities(module, init, pattern, cond, '+, 6) .
PD( {XOut | Ngf} {NgfRC | Ngf : NgfR} {TrkaC | _} {CLm | empty}
   {CLc | Ikk1 Ikk2 Irak1 Jnks Matk Myd88 Pkcd Ripk2 Sqstm1 Traf6
   Znf274} {CLi | [Hras - _] [Rap1a - _] [Rit1 - _]}  {NUc | _})
```

In these outputs, we find a ligand/receptor binding `NGF/NGFR`. In the search for depth 5 the proteins `Elk1` and `P38s` are present in the cytoplasm. However, in the search for depth 6, these proteins disappear as a common element. This results from the fact that a rule (`rl[618.P38s.irt.Ngf]`) is applied with activation of protein `P38s`.

4 Conclusions

Understanding of the dynamics of complex biological systems can be facilitated by developing symbolic methods. We formalize models that molecular biologists can use to think about signaling pathways and their behavior, allowing them to computationally formulate questions about their dynamics and outcomes. Rewriting logic gives us the ability to build and analyze models with multiple levels of detail; to represent biological rules; to define sorts of elements (chemicals, proteins, genes, locations, etc.) and their properties; and to precise queries using logical inference.

In this article we show an application of a rewriting logic procedure based in Maude logic language to the dynamic modeling of biological signaling pathways. As a case study, we analyze the role of nerve growth factor receptor signaling in cancer cell proliferation and survival using a reachability analysis approach. The characterization of neuron proliferation and survival is determined by protein activation of the families `AKT` and/or `ERK` [8]. We compute common elements in all possible solutions in our cases of interest. In conclusion, our results provide a reachability analysis in which a symbolic system explores complex and dynamic cellular signaling processes that induce cellular proliferation and cellular survival.

References

1. Clavel, M., Durán, F., Eker, S., Escobar, S., Lincoln, P., Martí-Oliet, N., Meseguer, J., Talcott, C.: Maude Manual (Version 2.7), March 2015. http://maude.cs.illinois.edu/w/images/1/1a/Maude-manual.pdf
2. Clavel, M., Durán, F., Eker, S., Lincoln, P., Martí-Oliet, N., Meseguer, J., Talcott, C.L.: All about maude - a high-performance logical framework, how to specify, program and verify systems in rewriting logic. In: LNCS, vol. 4350. Springer (2007)
3. Donaldson, R., Talcott, C.L., Knapp, M., Calder, M.: Understanding signalling networks as collections of signal transduction pathways. In: Quaglia, P. (ed.) Computational Methods in Systems Biology, CMSB, pp. 86–95. ACM (2010)
4. Gratie, D., Iancu, B., Petre, I.: ODE analysis of biological systems. In: Bernardo, M., de Vink, E.P., Pierro, A.D., Wiklicky, H. (eds.) 13th Int. School on Formal Methods for the Design of Computer, Communication, and Software Systems, SFM 2013. LNCS, vol. 7938, pp. 29–62. Springer (2013)
5. Hartman, D.S., McCormack, M., Schubenel, R., Hertel, C.: Multiple trkA proteins in PC12 cells bind NGF with a slow association rate. J. Biol. Chem. **267**(34), 24516–24522 (1992)

6. Martí-Oliet, N., Ölveczky, P.C., Talcott, C.L. (eds.) Logic, Rewriting, and Concurrency - Essays dedicated to José Meseguer on the occasion of his 65th birthday. LNCS, vol. 9200. Springer (2015)
7. Meseguer, J.: Twenty years of rewriting logic. J. Log Algebr. Program. **81**(7–8), 721–781 (2012)
8. Molloy, N.H., Read, D.E., Gorman, A.M.: Nerve growth factor in cancer cell death and survival. Cancers **3**(1), 510–530 (2011)
9. National Cancer Institute (US): Cancer Genome Anatomy Project. http://cgap.nci.nih.gov/Pathways (2016) (accessed February 29, 2016)
10. Santos-García, G., De Las Rivas, J., Talcott, C.L.: A logic computational framework to query dynamics on complex biological pathways. In: Adv. Intell. Syst. Comput., vol. 294, pp. 207–214. Springer (2014)
11. Santos-García, G., Talcott, C.L., De Las Rivas, J.: Analysis of cellular proliferation and survival signaling by using two ligand/receptor systems modeled by pathway logic. In: HSB 2015, pp. 226–245 (2015) (revised selected papers)
12. Talcott, C.L.: Pathway logic. In: Bernardo, M., Degano, P., Zavattaro, G. (eds.) Formal methods for computational systems biology. In: 8th Int. School on Formal Methods for the Design of Computer, Communication, and Software Systems, SFM 2008. LNCS, vol. 5016, pp. 21–53. Springer (2008)
13. Talcott, C.L., Eker, S., Knapp, M., Lincoln, P., Laderoute, K.: Pathway logic modeling of protein functional domains in signal transduction. In: Markstein, P., Xu, Y. (eds.) Proceedings of 2nd IEEE Computer Society Bioinformatics Conf, CSB 2003, Stanford, CA, August 11-14, 2003, pp. 618–619. IEEE Computer Society (2003)
14. Weng, G., Bhalla, U.S., Iyengar, R.: Complexity in biological signaling systems. Science **284**(5411), 92–96 (1999)

A Hybrid of Harmony Search and Minimization of Metabolic Adjustment for Optimization of Succinic Acid Production

Nor Syahirah Abdul Wahid, Mohd Saberi Mohamad, Abdul Hakim Mohamed Salleh, Safaai Deris, Weng Howe Chan, Sigeru Omatu, Juan Manuel Corchado, Muhammad Farhan Sjaugi, Zuwairie Ibrahim and Zulkifli Md. Yusof

Abstract Succinic acid has been favored by researchers due to its industrial multi-uses. However, the production of succinic acid is far below cell theoretical maximum. The goal of this research is to identify the optimal set of gene knockouts for obtaining high production of succinic acid in microorganisms. Gene knockout is a widely used genetic engineering technique. Hence, a hybrid of Harmony Search (HS) and Minimization of Metabolic Adjustment (MOMA) is proposed. The dataset applied is a core *Escherichia coli* metabolic network model.

N.S.A. Wahid · M.S. Mohamad(✉) · A.H.M. Salleh ·W.H. Chan
Artificial Intelligence and Bioinformatics Research Group, Faculty of Computing, Universiti Teknologi Malaysia, 81310 Skudai, Johor, Malaysia
e-mail: nsyahirah91@gmail.com, abdhakim.utm@gmail.com, saberi@utm.my, whchan2@live.utm.my

S. Deris
Faculty of Creative Technology and Heritage, Universiti Malaysia Kelantan, Locked Bag 01, 16300 Bachok, Kota Bharu, Kelantan, Malaysia
e-mail: safaai@umk.edu.my

S. Omatu
Department of Electronics, Information and Communication Engineering, Osaka Institute of Technology, Osaka 535-8585, Japan
e-mail: omatu@rsh.oit.ac.jp

J.M. Corchado
Biomedical Research Institute of Salamanca/BISITE Research Group, University of Salamanca, Salamanca, Spain
e-mail: corchado@usal.es

M.F. Sjaugi
Center for Bioinformatics, School of Data Sciences, Perdana University, 43400 Serdang, Selangor, Malaysia
e-mail: farhan@perdanauniversity.edu.my

M.S. Mohamad et al. (eds.), *10th International Conference on PACBB*, Advances in Intelligent Systems and Computing 477,
DOI: 10.1007/978-3-319-40126-3_19

Harmony Search is a meta-heuristic algorithm inspired by musicians' improvisation process. Minimization of Metabolic Adjustment is used to calculate fitness closest to the wild-type, after mutant gene knockout. The result obtained from the proposed hybrid technique are knockout genes list and production rate after the deletion. This proposed technique is possible to be applied in wet laboratory experiment to increase the production of succinic acid in *E. coli*.

Keywords Bioinformatics · Artificial intelligence · Metabolic engineering · Harmony Search · Minimization of Metabolic Adjustment · Gene knockout

1 Introduction

Succinic acid and its derivatives have been widely used in various industrial applications such as food additives, flavoring agents, pharmaceutical supplements, surfactants, detergent extenders, foaming agents, and ion chelators [1]. However, the production of succinic acid using traditional methods (glucose medium fermentation of microbial strains in aerobic and anaerobic conditions) is far below its theoretical maximum. Moreover, it is time consuming and costly. Gene knockout is a genetic engineering technique to inactivate gene of interest in an organism, which lead to deletion of a particular protein and cause changes in phenotype and affect the biochemical production. Several computational methods have been developed to identify gene knockout strategy. OptKnock [2] is one of the methods, which suggests gene deletion strategies that lead to the improved biochemical production in E. coli. However, the problem of finding optimal gene deletion strategy in OptKnock is combinatorial and leads to the high computational time [3]. OptGene [4] is an extension of OptKnock, which use Genetic Algorithm (GA) to solve larger problem with reduced computational time. However, in OptGene, the number of invalid individuals in the population in a given generation is very high and this negatively affect the convergence rate. Moreover, GA can also cause premature convergence. Another method is MOMAKnock [5], which is a new bi-level optimization framework that aims to maximize the production of targeted chemicals by identifying candidate knockout genes or reactions under phenotypes constraints calculated by the MOMA assumption.

MOMA is a flux-based analysis method based on stoichiometric constraints. MOMA is proposed by Segre et al. (2002), which provides a mathematically tractable estimation for the state in between wild type, optimum and mutant

Z. Ibrahim
Faculty of Electrical and Electronics Engineering, Universiti Malaysia Pahang, 26600 Pekan, Pahang, Malaysia
e-mail: zuwairie@ump.edu.my

Z.Md. Yusof
Faculty of Manufacturing Engineering, Universiti Malaysia Pahang, 26600 Pekan, Pahang, Malaysia
e-mail: zmdyusof@ump.edu.my

optimum, and assumes that the mutant remains as close as possible to the wild-type optimum in terms of flux values [6]. MOMA is used to predict metabolic flux distributions at steady state in a perturbed system. It identifies a point in flux space, which is closest to the wild-type point, compatible with the gene deletion constraint. MOMA can serve as a more precise method for predicting the metabolic phenotype of gene knockout. Furthermore, MOMA can be applied to predict the metabolic phenotype of gene knockout. However, predicting the optimal set of gene to be knocked out in order to achieve the optimal production of chemical is a limitation in MOMA. Meanwhile, Harmony Search (HS), which is a meta-heuristic optimization algorithm can be applied to solve this problem. HS algorithm is proposed by Geem et al. [7] is based on searching the perfect state of harmony in a musical process. The HS algorithm does not require initial value and it uses a random search instead of gradient search, hence derivative information is not required [7].

In this paper, a hybrid of HS and MOMA is proposed, known as HSMOMA. The goal of this research is to identify the optimal set of gene knockouts for obtaining high production of succinic acid in microorganisms. HS is used to identify the optimal set of genes to be knockout whereas MOMA is used as a fitness function to simulate the phenotype behavior of *E. coli* after gene knockout.

2 Method

2.1 Harmony Search Algorithm

Harmony Search is an algorithm inspired by the improvisation process of musicians [7]. Figure 1 shows the pseudo code of HS algorithm. In HS algorithm, a decision variable selects one value for improvisation based on one of the three rules: (1) choosing any one value from the HS memory (referred as memory considerations), (2) choosing an adjacent value of one value from the HS memory (referred as pitch adjustments), and (3) choosing a totally random value from the possible range of value (referred as randomization) [8]. Two parameters in HS include harmony memory considering the rate (HMCR) and pitch adjusting rate (PAR).

```
Harmony Search
  begin
      Objective function f(x), x=(x1,x2, ...,xd)^T
       Generate initial harmonics (real number arrays)
      Define pitch adjusting rate (PAR), pitch limits and bandwidth
      Define harmony memory considering rate (HMCR)
      while (t < Max number of iterations )
            Generate new harmonics by accepting best harmonics
            Adjust pitch to get new harmonics (solutions)
            if (rand > HMCR), choose an existing harmonic randomly
            else if (rand > PAR), adjust the pitch randomly within limits
            else generate new harmonics via randomization
            end if
            Accept the new harmonics (solutions) if better
      end while
      Find the current best solutions
  end
```

Fig. 1 Harmony Search Algorithm Pseudo code [9].

2.2 Minimization of Metabolic Adjustment

MOMA employs quadratic programming (QP) to find the flux distribution which has the smallest Euclidean distance to the wild type situation after mutations occurred [10]. The aim is to find the vector x $\in \Phi^j$, where Φ means of feasible space.

$$\mathrm{D}\ (v^w, v^m) = \sqrt{\sum_{i=1}^{N}(v_i^m - v_i^w)^2} \tag{1}$$

where v_i^w = flux of the wild type for the i-th reaction and, v_i^m = calculation of flux of the mutants for the i-th reaction. The equation could be transformed to the standard form for QP.

$$fx=L.x+12x^T Qx \tag{2}$$

where vector *L* = length *N*, matrix *Q* = *N***N* matrix, define the linear and quadratic part of the objective function, respectively. x^T is the transpose of *x*. Function D of Equation 1 is equal to minimizing its square, and the constant terms can be omitted from the objective function, once can choose Q to be N×N matrix and set L = -w, hence reduce the minimization of D to the minimization of f(x). Its linear part implies that the vector of fluxes of the wild type.

2.3 A Hybrid of Harmony Search and Minimization of Metabolic Adjustment

This paper proposes a hybrid of Harmony Search algorithm and Minimization of Metabolic Adjustment (HSMOMA). The flowchart is as shown in Figure 2.

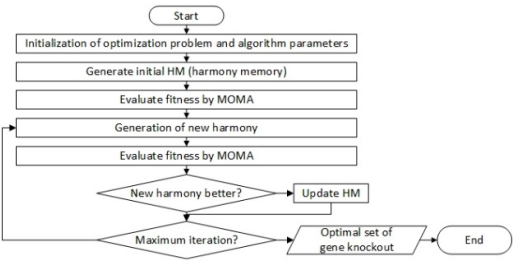


Fig. 2 Flowchart of Hybrid Algorithm of HSMOMA

2.3.1 Initialization of Optimization Problem and HS Algorithm Parameters

Firstly, the objective function *f(x)* which is subjected to the constraint functions and algorithm parameters are defined. The parameters that are required to solve the objective function such as harmony memory size (number of solution vectors,

HMS), harmony memory considering the rate (HMCR), pitch adjusting rate (PAR) and termination criterion (maximum number of searches) are specified in this phase as well.

2.3.2 Initialization of Harmony Memory (HM)

In this phase, the HM matrix of *m* x *n* is randomly generated (with the values of 0 and 1) where *m* represents the number of reactions in *E. coli* and *n* represents the number of populations. 0 indicates that the reaction cannot be knocked out while 1 indicates that the reaction can be knocked out in order to optimize the metabolite production. Once the HM is initialized, the list of reactions that can be knocked out become the input of MOMA for calculating the objective function, which is the production of metabolites in *E. coli* after the knockout of reaction.

2.3.3 Generate a New Harmony

In this phase, a new harmony vector, $(x' = x'_1, \ldots, x'_N)$ is generated from HM or possible variable values based on rules explained in Section 2.1. HMCR determines the rate of choosing any value from the historic values stored in HM and (1-HMCR) determines the rate of choosing one value from an entire feasible range. Next, HS algorithm examines each component of the new harmony vector to determine whether it should be pitch-adjusted by using PAR parameter. Both HMCR and PAR parameters enable HS to search for optimal solution locally and globally. MOMA is then used to calculate the fitness function of the new harmony.

2.3.4 Update HM

HM is updated if there is a better harmony vector based on the fitness function value calculated by MOMA. The new harmony is then included in the HM and the worse one is excluded. Finally, the HM is sorted based on the fitness function value.

2.3.5 Fitness Function

The fitness function used in this paper is production yield, which is the maximum amount of product that can be generated per unit of substrate. The formula of the fitness function is shown in Equation 3.

$$production\ yield = \frac{production\ rate\ \ (mmol\ \mathrm{gDW^{-1}hr^{-1}})}{consumption\ rate\ substrate\ (mmol\ \mathrm{gDW^{-1}hr^{-1}})}$$

2.3.6 Termination

After the maximum number of iterations, the algorithm is terminated. The best solution for reaction list needs to be knockout is generated.

3 Result and Discussion

A core *E. coli* metabolic network model obtained from literature [11] is used in this research, which consists of 137 genes, 95 reactions, and 72 metabolites. The HMCR and PAR parameters are set to 0.95 and 0.3 respectively. Besides, the glucose uptake rate was fixed to 10 mmol gDW^{-1} hr^{-1}. Next, HSMOMA is executed for 50 runs (100 iterations in each run) in order to achieve the highest growth rate and succinic acid production rate. HSMOMA is tested by setting the number of knockout (K) from 1 to 5. The growth rate of *E. coli* is measured in unit one per hour (hr^{-1}) and 0.1 is set as the minimum growth rate to ensure that the cell is alive and every population with less than growth rate is removed. While, the production rate of succinic acid is measured in units millimole per gram dry cell weight per hour (mmol $gDW^{-1}hr^{-1}$). Figure 3 shows the maximum production of succinic acid for every K.

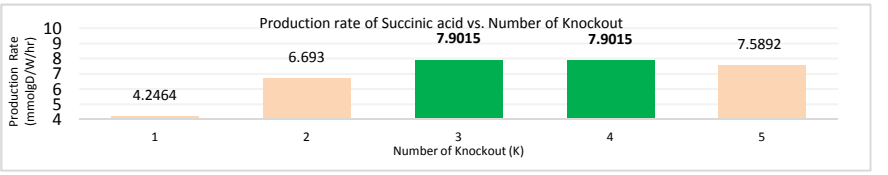


Fig. 3 Maximum Production of Succinic Acid (50 runs of HSMOMA)

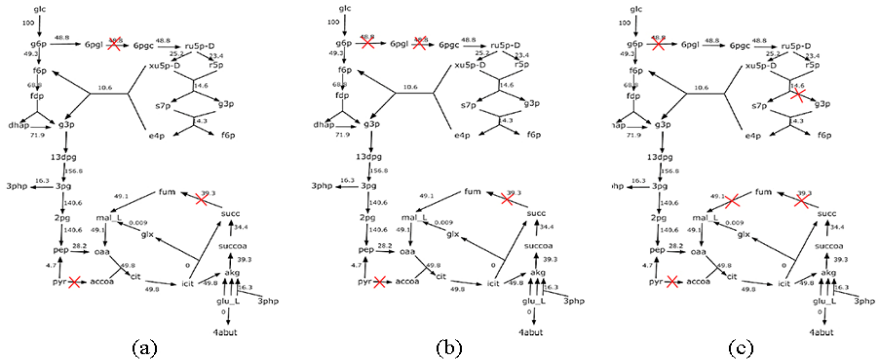


Fig. 4 Types of Mutant: (a) Mutant A (b) Mutant B (c) Mutant C

The results of hybrid algorithm of HSMOMA is presented in Table 1. There are three sets of knockout list that have the best production of succinic acid. Both mutant A and mutant B (Figures 4(a) and 4(b)) show the highest production rate of succinic acid, which is 7.9015 mmol $gDW^{-1}hr^{-1}$ and the growth rate is 0.2761hr^{-1}. Based on Table 1, HSMOMA suggests knockout reactions that consume succinic acid such as pyruvate dehydrogenase (PDH), 6-phosphogluconolactonase (PGL), succinate dehydrogenase (SUCDi), and glucose 6-phosphate dehydrogenase (G6PDH2r).

Table 1 Results for Different Set of Gene Knockout for Succinic Acid

Mutant	Enzyme	Gene	Succinic Acid Production Rate (mmol $\mathbf{gDW^{-1}hr^{-1}}$)	GrowthRate ($\mathbf{hr^{-1}}$)
A	PDH PGL SUCDi	*aceE, aceF, lpdA* pgl sdh	7.9015	0.2761
B	PDH PGL SUCDi G6PDH2r	*aceE, aceF, lpdA* pgl sdh *zwf*	7.9015	0.2761
C	PDH G6PDH2r SUCDi FUM TKT1	*aceE, aceF, lpdA* *zwf* *fumAB* *tktA, tktB*	7.5892	0.31356

Note: The highlighted section in Table 1 shows the highest production rate of succinic acid.

Table 2 Comparison of the results obtained from OptKnock, MOMAKnock, and HSMOMA.

Method	Knockouts	Enzyme	Production Rate (mmol $\mathbf{gDW^{-1}hr^{-1}}$)
Opt-Knock	COA + PYR → ACOOA + FOR NADH + PYR ↔ LAC + NAD	Pyruvate formate lyase (PFL) Lactate dehydrogenase (LDH_D)	1.65
	COA + PYR → ACOOA + FOR NADH + PYR ↔ LAC + NAD ACCOA + 2 NADH ↔ COA + ETH + 2 NAD	Pyruvate formate lyase (PFL) Lactate dehydrogenase (LDH_D) Acetaldehyde dehydrogenase (ACALD)	4.79
	ADP + PEP → ATP + PYR ACTP + ADP ↔ AC + ATP or ACCOA + Pi ↔ ACTP + COA GLC + PEP → G6P + PYR	Pyruvate kinase (PYK) Acetate kinase (ACKr) Phosphotransacetylase (PTAr) Phosphotransferase system (GLCpts)	6.21
MOMA-Knock	Q8 + SUCC → FUM + Q8H2 6PGL + H2O → 6PGC + H (2)H2O + O2 + URATE → ALLTN + CO2 + H2O2	Succinate dehydrogenase (SUCDi) 6-phosphogluconolactonase (PGL) Uricase (URIC)	5.02
	Q8 + SUCC → FUM + Q8H2 AC + ATP → ACTP + ADP H2O + METHF → 10FTHF+ H 5P + XU5P-D → G3P + S7P	SUCDi ACKr Methylenetetrahydrofolate dehydrogenase (MTHFD) TKT1	5.02
	Q8 + SUCC → FUM + Q8H2 GLU-L+H → 4ABUT + CO2 3PG + NAD → 3PHP + H + NADH 3PHP + GLU-L → AKG + PSERL 6PGC + NADP → CO2 + NADPH + RU5P-D	SUCDi Glutamate Decarboxylase (GLUDC) Phosphoglycerate dehydrogenase (PGCD) Phosphoserine transaminase (PSERT) Phosphogluconate dehydrogenase (GND)	5.02
HS-MOMA	COA + NAD + PYR → ACCOA + CO2 + NADH 6PGL + H2O → 6PGC + H Q8 + SUCC → FUM + Q8H2	PDH PGL SUCDi	7.9015
	COA + NAD + PYR → ACCOA + CO2 + NADH 6PGL + H2O → 6PGC + H Q8 + SUCC → FUM + Q8H2 G6P + NADP ↔ 6PGL + H + NADPH	PDH PGL SUCDi G6PDH2r	7.9015
	COA + NAD + PYR → ACCOA + CO2 + NADH G6P + NADP ↔ 6PGL + H + NADPH Q8 + SUCC → FUM + Q8H2 FUM + H2O ↔ MAL-L 5P + XU5P-D ↔ G3P + S7P	PDH G6PDH2r SUCDi FUM TKT1	7.5892

Note: The highlighted section in Table 2 shows the highest production rate of succinic acid

The deletion of PDH causes NADH to be fully utilized in tricarboxylic acid cycle (TCA cycle) because the production of acetate and ethanol are confined [12]. Hence, the production of succinic acid is increased. While deletion of SUCDi also increases the succinic acid production because it catalyzes the reaction that convert

succinic acid to fumarate [13]. In previous literature, the deletion of G6PDH2r also increases the production rate of succinic acid because it catalyzes the reaction that reduces succinic acid [14]. However, in this research shows *zwf* does not really effect the succinic production in mutant B. In mutant C (Figure 4(c)), the knockout reactions are fumarase (FUM), and transketolase (TKT1) enzymes. FUM catalyzes the reaction between fumarate and malate, which do not affect production of succinic acid [8]. Meanwhile, the knockout of TKT1 helps to increase the production of succinic acid because it directly consumes succinic acid [5]. Table 2 shows the comparison of the results between HSMOMA, OptKnock and MOMAKnock in terms of production rate of succinic acid by using the same dataset. In previous literature, OptGene only calculate growth rate but this research is focusing on the production rate value. The result obtained from HSMOMA is better than previous studies. This shows that HSMOMA is able to identify the optimal set of gene knockouts in optimizing the production of succinic acid.

4 Conclusion and Future Works

In this research, we propose a hybrid algorithm known as HSMOMA to identify the gene knockout strategies for optimizing the production of succinic acid in *E. coli*. The result shows that the hybrid HSMOMA performs better than the other related research and has successfully identified three sets of gene knockout that optimize the production of succinic acid. For future works, we suggest some modifications in HS to improve the balance between exploration and exploitation process in harmony memory. Moreover, kinetic and regulatory information can be added to MOMA to better calculate the flux balance in a cell.

Acknowledgements We would like to thank Universiti Teknologi Malaysia for funding this research through the Research University Grant (Grant number: Q.J130000.2528. 12H12). This research also supported by the Malaysian Ministry of Education through the Fundamental Research Grant Schemes (Grant numbers: R.J130000.7828.4F720, RDU140114, and RDU140132).

References

1. Zeikus, J.G., Jain, M.K., Elankovan, P.: Biotechnology of Succinic Acid Production and Markets for Derived Industrial Products. Applied Microbiology and Biotechnology **51**, 545–552 (1999)
2. Burgard, A.P., Pharkya, P., Maranas, C.D.: Optknock: A Bilevel Programming Framework for Identifying Gene Knockout Strategies for Microbial Strain Optimization. Biotechnology and Bioengineering **84**, 647–657 (2003)
3. Rocha, I., Maia, P., Rocha, M., Ferreira, E.C.: OptGene: a framework for in silico metabolic engineering. In: 10th International Conference on Chemical and Biological Engineering, pp. 218–219. University of Minho, Portugal (2008)
4. Patil, K.R., Rocha, I., Förster, J., Nielsen, J.: Evolutionary Programming as A Platform for In Silico Metabolic Engineering. BMC Bioinformatics **6**, 308 (2005)

5. Ren, S., Zeng, B., Qian, X.: Adaptive Bi-Level Programming for Optimal Gene Knockouts for Targeted Overproduction Under Phenotypic Constraints. BMC Bioinformatics **14**, S17 (2013)
6. Segre, D., Vitkup, D., Church, G.M.: Analysis of Optimality In Natural and Perturbed Metabolic Networks. Proceedings of the National Academy of Sciences **99**, 15112–15117 (2002)
7. Geem, Z.W., Kim, J.H., Loganathan, G.V.: A New Heuristic Optimization Algorithm: Harmony Search. Simulation **76**, 60–68 (2001)
8. Lee, K.S., Geem, Z.W.: A New Meta-Heuristic Algorithm for Continuous Engineering Optimization: Harmony Search Theory and Practice. Computer Methods In Applied Mechanics and Engineering **194**, 3902–3933 (2005)
9. Yang, X.S.: Harmony Search as a metaheuristic algorithm through web services. In: Zong, W.G. (ed.) Music-inspired Harmony Search Algorithm, vol. 191, pp. 1–14. Springer, Heidelberg (2006)
10. Papp, B., Pal, C., Hurst, L.D.: Metabolic Network Analysis of The Causes and Evolution of Enzyme Dispensability In Yeast. Nature **429**, 661–664 (2004)
11. Orth, J.D., Fleming, R.M., Palsson, B.Ø.: Reconstruction and use of microbial metabolic networks: the core Escherichia coli metabolic model as an educational guide. EcoSal Plus **4**(1) (2010)
12. Vemuri, G.N., Eiteman, M.A., Altman, E.: Succinate Production In Dual – Phase Escherichia Coli Fermentations Depends on The Time of Transition from Aerobic to Anaerobic Conditions. Journal of Industrial Microbiology & Biotechnology **28**, 325–332 (2002)
13. Lee, S.J., Lee, D.Y., Kim, T.Y., Kim, B.H., Lee, J., Lee, S.Y.: Metabolic Engineering of Escherichia Coli for Enhanced Production of Succinic Acid, Based on Genome Comparison and In Silico Gene Knockout Simulation. Applied and Environmental Microbiology **71**, 7880–7887 (2005)
14. Zimenkov, D., Gulevich, A., Skorokhodova, A., Biriukova, I., Kozlov, Y., Mashko, S.: Escherichia Coli ORF ybhE is pgl Gene Encoding 6-Phosphogluconolactonase (EC 3.1.1.31) That Has No Homology with Known 6pgls From Other Organisms. FEMS Microbiology Letters **244**, 275–280 (2005)

Development of an Integrated Framework for Minimal Cut Set Enumeration in Constraint-Based Models

Vítor Vieira, Paulo Maia, Isabel Rocha and Miguel Rocha

Abstract Under the realm of *in silico* Metabolic Engineering, pathway analysis approaches to strain optimization have shown a large potential as tools capable of providing an unbiased view over metabolic models. Most of these methods were difficult or impossible to use due to their heavy computational needs, since they are based in the calculation of elementary modes/minimal cut sets in large networks. However, a recent method (*MCSEnumerator*) has enabled the application of these approaches to genome-scale metabolic models. This work proposes a new software tool where this method is implemented in a novel Java library, that provides support for a plugin for the OptFlux metabolic engineering platform. Together, these tools implement the routines necessary for the calculation of minimal cut sets and their use to provide strain optimization methods. The aim is to provide an open-source software tool that includes an intuitive graphical user interface, thus facilitating its use by the community.

Keywords Metabolic pathway analysis · Constraint-based model · Strain optimization · Minimal cut sets · Metabolic engineering

1 Introduction

In recent years, genome-scale metabolic models (GSMMs) encompassing annotated whole genomes of living organisms have proved useful in predicting cell phenotypes through *in silico* methods. A particularly interesting application of GSMMs concerns the field of metabolic engineering (ME) which aims to design enhanced cell factories

V. Vieira(✉) · I. Rocha · M. Rocha
Centre of Biological Engineering, University of Minho, Braga, Portugal
e-mail: jose.vieira153@gmail.com

P. Maia
SilicoLife Lda., Braga, Portugal

M.S. Mohamad et al. (eds.), *10th International Conference on PACBB*,
Advances in Intelligent Systems and Computing 477,
DOI: 10.1007/978-3-319-40126-3_20

for products with added value in industrial biotechnology. The use of metabolic models allows for a rational design process, integrating vast amounts of data instead of trial-and-error methodologies [1, 2]. Most methods based on the use of GSMMs follow a constraint-based (CB) approach, considering various assumptions regarding cell metabolite balancing and discarding enzyme kinetics, as this information is only partially available. Various phenotype prediction, analysis and strain optimization methods have been developed using this approach [3]. Phenotype prediction methods are based on mathematical formulations that assume cell metabolism is driven towards certain goals. The methods developed so far include variants for wild-type strains, the most popular being Flux Balance Analysis (FBA) [4], and for mutant strains such as Minimization of Metabolic Adjustment (MOMA) [5] and Regulatory On/Off Minimization (ROOM) [6].

Computational strain optimization methods (CSOMs), with the purpose of finding genetic manipulation strategies able to overproduce selected compounds, have also been developed following the CB approach [7]. In this work, only optimization methods involving reaction deletions will be approached. These can be divided in two broad categories: bi-level constraint-based methods and pathway analysis (PA) methods. Bi-level approaches are nested optimization problems attempting to find engineering strategies that increase product yields or titers, as well as optimizing cellular objectives, through phenotype prediction methods, mostly FBA and related methods as MOMA or ROOM. While such assumptions allow faster computational time, assuming an objective may lead to bias which can result in less robust strategies. Some of these methods use deterministic methods, such as OptKnock [8], while others employ stochastic meta-heuristics, as Evolutionary Computation (e.g. OptGene [9]).

Pathway analysis approaches only consider the metabolite balance assumption, decreasing bias. Most are based on elementary modes (EMs), a concept representative of basic cell functions contained in a model. Complete enumeration of all EMs, and in some cases the related Minimal Cut Sets (MCSs), in a network is necessary for most PA methods, but incurs in heavy computational demands. However, EMs and MCSs [10] are important assets in strain optimization. Also, recent methods have allowed to extend the computation of EMs and MCSs to metabolic models at a genome-wide scale, through the MCSEnumerator framework [11].

In this scenario, given the potential of MCSs to guarantee robust production, regardless of the phenotype prediction method used, this work pursues the development of an open-source software tool capable of handling relevant tasks associated with their enumeration and applying those to strain optimization. Therefore, the main scientific/technological objectives of this work are:

- to implement a library containing the necessary routines for enumeration of MCSs in metabolic networks;
- to integrate MCS enumeration tasks in the OptFlux metabolic engineering platform, providing novel tools for strain optimization;
- to provide a simple and intuitive user-interface for the implemented routines.

2 Methods

2.1 Constraint-Based Models and Pathway Analysis

Constraint-based models of metabolism comprise m intracellular metabolites and n reactions acting upon them. These reactions also include sinks for external metabolites, representing their uptake and/or production. The system is represented by a $m \times n$ matrix S, containing stoichiometric coefficients. In CB methods, metabolite concentrations are assumed to be time invariant, leading to a system of linear equations:

$$S \cdot v^T = 0 \tag{1}$$

with v as the column vector of fluxes (or rates) for each individual reaction. Additionally, thermodynamics assumptions and/or rate limits are added as additional constraints in the form:

$$\alpha \leq v \leq \beta \tag{2}$$

with α and β being respectively the vectors containing lower and upper limits for each element (flux) in v. Any irreversible reaction j must have a lower limit $\alpha_j = 0$. The system defined by Equations 1 and 2 can also be represented in space as a convex polyhedron hereby referred to as P, containing all feasible solutions to this system.

Considering this modeling framework, an **elementary mode** (EM) represents the smallest functional unit within it. Any elementary mode e equates to a flux distribution obeying three key properties [12]:

1. A flux distribution in e must comply with Equation 1;
2. Irreversible reactions must carry flux only through a single direction in any EM. These are specified in Equation 2;
3. Considering **supp**(e) as the reactions carrying flux in e, no subset of $supp(e)$ can yield a flux distribution obeying Equations 1 and 2.

Any point contained within P can be defined as a linear combination of EMs. It is possible to find desirable solutions to the metabolic model by finding points described by non-null combinations of EMs contained within a desired set of flux vectors D. Conversely, any set of undesired flux vectors T can be blocked by disabling EMs contained within that space. A set of reactions blocking all vectors in T is a cut set of T. If no reaction can be removed from the cut set without rendering it unable to block the vectors in T, it is considered a **minimal cut set** (MCS). However, MCSs do not necessarily guarantee the set of desired vectors D will not be blocked as well. An MCS M is considered a **constrained minimal cut set** (cMCS) [13] if it blocks all EMs describing the space in T, as well as ensuring points in D are feasible solutions to the system.

2.2 Enumerating Minimal Cut Sets

Most methods for the enumeration of MCSs involve prior knowledge of the full set of EMs in the network and are usually based on combinatorial algorithms. However, these are unsuitable for GSMMs due to the heavy computational demand of this task. Problem complexity and the number of total EMs rise exponentially with the size of the model, rendering it virtually impossible.

However, a recent approach, MCSEnumerator, has been proposed which is capable of enumerating MCSs and cMCSs in GSMMs, in some cases up to seven knockouts [11]. This algorithm involves a mixed-integer linear programming problem (MILP) that allows partial enumeration of MCSs with good time-efficiency. This formulation derives from a finding documented in [14] and describing the formulation of a dual system in which EMs correspond to MCSs in the original metabolic model. This is currently the most suitable approach for enumeration of MCSs in GSMMs, which prompted its use in this work.

A generic pipeline based on the original publication was assembled, covering all required steps for the enumeration task, as shown in the left panel of Figure 1. The model compression step in the pre-processing phase and subsequent MCS decompression in the enumeration phase are optional, but speed up computation times. The MILP framework developed in [11] as well as a basic algorithm for MCS enumeration are represented on the right side of the same figure. These constraints, along with the dual system, result in an EM enumeration problem (using the k-shortest EFM algorithm [15]) where EMs represent MCSs in the original network.

3 Development

This work has two main outcomes regarding the developed software. The first is a *Java* library implementing an entire pipeline for MCS enumeration using the algorithm in [11] and their use for strain optimization. The second outcome is a plugin developed for the OptFlux platform, providing a user interface for the library.

3.1 Enumeration Library

MCSEnumerator is currently only available as a part of the CellNetAnalyzer platform for MATLAB. One of the aims of this work was to build an independent library containing the necessary routines and algorithms for MCS enumeration using the MCSEnumerator [11] approach. This library was built using the *Java* programming language, allowing greater portability, as well as enabling the use of more advanced tools for the development of a graphical user interface (GUI). Currently, it requires

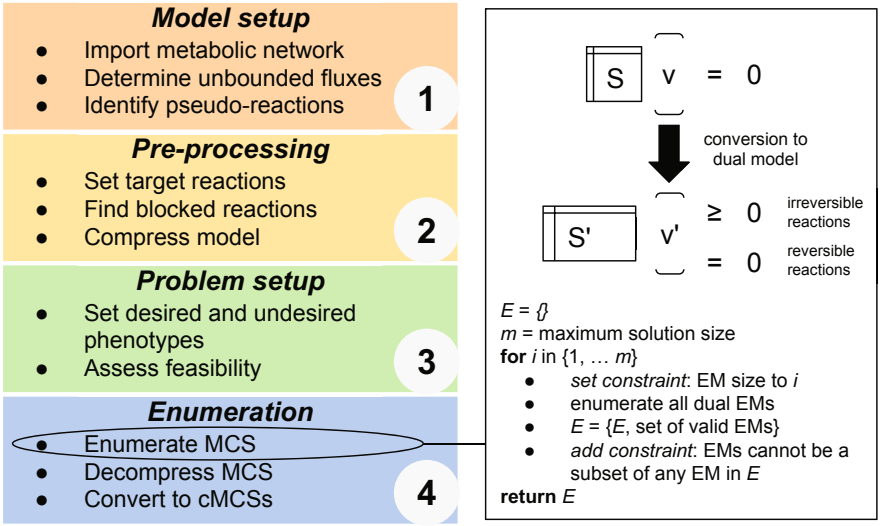


Fig. 1 **Left:** Representation of the pipeline used in this work. Step **(1)** concerns model setup which consists mainly on determining reaction reversibility and pseudo-reactions that will not be a part of any solution; Step **(2)** aims at reducing the size of the problem, mostly through removal of blocked reactions and network compression by lumping correlated reactions. Step **(3)** assembles the enumeration problem and validates it, so that in Step **(4)** the proper formulation is built and solved. MCSs that are not feasible in the desired space are discarded, leaving only cMCSs. **Right:** Brief overview of the MILP formulation. S' and v' are derived from the dual model formulation in [14] which already includes the undesired phenotypes.

the usage of the IBM ILOG CPLEX Optimization Studio[1] for solving the MILP problem described in [11].

This library contains three newly developed packages:

1. **Enumeration**: Contains methods needed to implement the MCSEnumerator MILP formulation, given a suitable problem.
2. **Metabolic**: Provides a framework upon which constraint-based metabolic models can be defined, as well as constraints for typical linear programming problems such as yield or capacity constraints on the reactions.
3. **Utilities**: Includes methods that execute the entire pipeline given a set of parameters for the optimization. Functions to run the algorithms in a command-line environment are also provided, capable of reading parameters contained within text files.

The libraries provide routines capable of executing these tasks in a command-line environment using only a metabolic model in Systems Biology Markup Language (SBML) format and a file containing the parameters of the MCS enumeration problem, which may include undesired or desired limits for fluxes or yields, exclusion of target reactions, among other constraints.

[1] http://www-03.ibm.com/software/products/en/ibmilogcpleoptistud

3.2 OptFlux Plugin

This work also aimed at providing a simple and clear user interface (GUI) as a plugin within OptFlux [16]. Currently, this framework includes important tools used in CB approaches, including phenotype simulation, analysis methods, and strain optimization algorithms developed in-house (OptGene[9] and derivatives [17]).

As far as this work is concerned, OptFlux provides the necessary methods to read and write metabolic models, serving as inputs for our algorithms. The developed plugin provides a simple GUI, shown in Figure 2 for the MCSEnumerator approach requiring minimal user input and providing a useful abstraction for the concepts discussed in the previous section. The user is only required to specify the maximum number of knockouts, which reactions correspond to biomass, product synthesis and substrate uptake, the desired thresholds for production and growth, and whether the production threshold is a yield or a rate constraint. Additionally, environmental conditions can be added, and knockout targets can be discarded from the search either by supplying a list of critical reactions or a gene ID corresponding to spontaneous reactions, should the model represent those as being associated with a placeholder pseudo-gene. The solutions are displayed using OptFlux's GUIs, using the format of previously available optimization algorithms. So, these solutions can be processed and simulated afterwards using other tools from OptFlux. From OptFlux 3.3 onwards, this plugin is available in the software's plugin repository.

4 Results

4.1 Library Validation

The set of case studies was defined with the aim of ensuring that the outputs provided by the developed software match the ones from MCSEnumerator's original implementation. As such, the iAF1260 *Escherichia coli* GSMM [18] was used with different enumeration problems for which the cMCSs were previously determined in the original publication [11]. The results from all case studies were accurately replicated and are highlighted in the Table 1.

4.2 Plugin Operation

This section shows in more detail the plugin's mode of operation using one of the case studies described above (anaerobic ethanol production in *Escherichia coli* using glucose as carbon source). To run this case study:

Table 1 Overview of the validation case studies. Y represents product/substrate yield and Glc represents glucose uptake ($mmol \cdot gDW^{-1} \cdot h^{-1}$). Note that aerobic conditions were allowed only for fumarate and serine production. Computation times were determined in a single run using 12 cores (from two Intel® Xeon® E5-2650 CPUs) and 30GB of RAM.

Objective	Scenario	#MCS/#cMCS	Computation time (h)	Maximum size
Synthetic lethals	-	1018 / -	17	4
Anaerobic ethanol production	Glc $\leq$ 10 Y $\geq$ 1.4	185302 / 8342	7.5	7
	Glc $\leq$ 10 Y $\geq$ 1.8	153338 / 1987	9.1	7
	Glc $\leq$ 18.5 Y $\geq$ 1.4	156477 / 8819	12.7	7
	Glc $\leq$ 18.5 Y $\geq$ 1.8	138675 / 4618	2	7
Fumarate production	Y $\geq$ 0.5	17338 / 30	12.4	7
Serine production	Glc $\leq$ 20	18449 / 140	1	6

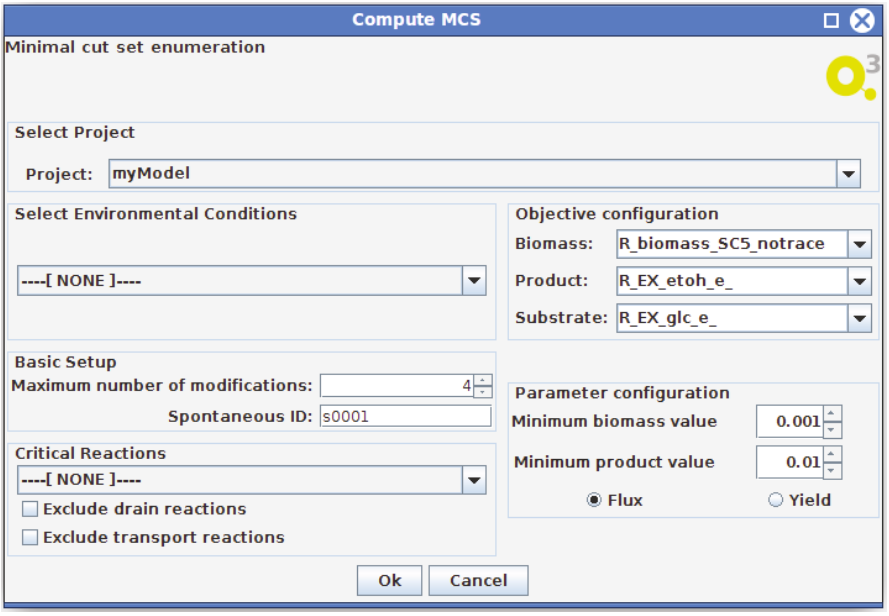


Fig. 2 Graphical interface provided by the plugin to formulate an enumeration problem.

1. Start a new OptFlux project using the **New project wizard** option, click on **OptFlux model repository** and select the iAF1260 *Escherichia coli* model. Assume default options in the process.
2. Create an environmental condition using the **New...** menu, option **Create...** and then click **Environmental condition**. Add a constraint for reaction *R_EX_glc_e* with lower bound as -20 and upper bound as 999999 (definition of glucose uptake

rate), and another for *R_EX_o2_e_* with 0 as lower bound and 999999 as upper bound (definition of anaerobic conditions).

3. Access the **Optimization** tab, and click on **Minimal cut sets**.
   a. Select the environmental condition that was created in the previous step.
   b. Allow at most 3 modifications and set the spontaneous ID for s0001.
   c. Configure the objectives as follows: Biomass as *R_Ec_biomass_core_59p81M*, substrate as *R_EX_glc_e_* and product as *R_EX_etoh_e_*.
   d. Set the biomass value to 0.1, choose yield and set the minimum product value to 0.2.

This example allows determination of up to 3 knockouts guaranteeing a production yield of at least 0.2 with a growth rate of $0.1h^{-1}$. The results can be browsed and sorted and also saved to disk as a text file. Specific solutions (deletion sets) can be saved to the clipboard, and simulated or analyzed through other OptFlux tools.

5 Conclusions and Further Work

The availability of PA-based strain design methods is scarce when considering GSMMs. The new library proposed in this work presents a useful resource for the metabolic engineering community, allowing for the enumeration of MCSs, in a way that is standardized fit for most problems with generic CB models, while also allowing flexibility regarding problem setup. The proposed OptFlux plugin facilitates an abstraction from complex concepts surrounding cMCS enumeration, improving ease of use and extending the already wide variety of optimization algorithms within OptFlux, maintaining a coherent overall computational interface. Also, the provided software is all made available to the community as open source allowing for third party contributions in the future. Despite all efforts, the library is still dependent on a commercial solver, but this provides a free academic license.

Computation times for larger sets of deletions, even when using a state-of-the-art optimizer, can be time consuming for some enumeration problems, and others still remain out of reach. Heuristic methods or alternative formulations may help in achieving solutions for larger sizes and this will be a line of future work.

Acknowledgments The authors thank the project "DeYeastLibrary - Designer yeast strain library optimized for metabolic engineering applications", Ref. ERA-IB-2/0003/2013, funded by national funds through FCT/MCTES.

References

1. Stephanopoulos, G.: Metabolic Fluxes and Metabolic Engineering 11 (1999)
2. Patil, K.R., Åkesson, M., Nielsen, J.: Use of genome-scale microbial models for metabolic engineering. Curr. Opin. Biotechnol. **15**(1), 64–69 (2004)

3. Szallasi, Z., Stelling, J., Periwal, V.: System Modeling in Cell Biology (2010)
4. Varma, A., Palsson, B.O., Arbor, A., Varma, A.: Stoichiometric Flux Balance Models Quantitatively Predict. Appl. Environ. Microbiol. **60**(10), 3724–3731 (1994)
5. Segrè, D., Vitkup, D., Church, G.M.: Analysis of optimality in natural and perturbed metabolic networks. Proc. Natl. Acad. Sci. U. S. A. **99**(23), 15112–15117 (2002)
6. Shlomi, T., Berkman, O., Ruppin, E.: Regulatory on/off minimization of metabolic flux changes after genetic perturbations. Proc. Natl. Acad. Sci. U. S. A. **102**(21), 7695–7700 (2005)
7. Maia, P., Rocha, M., Rocha, I.: In silico constraint-based strain optimization methods: the quest for optimal cell factories. Microbiology and Molecular Biology Reviews **80**(1), 45–67 (2016)
8. Burgard, A.P., Pharkya, P., Maranas, C.D.: Optknock: A bilevel programming framework for identifying gene knockout strategies for microbial strain optimization. Biotechnol. Bioeng. **84**(6), 647–657 (2003)
9. Patil, K.R., Rocha, I., Förster, J., Nielsen, J.: Evolutionary programming as a platform for in silico metabolic engineering. BMC Bioinformatics **6**(1), 308 (2005)
10. Klamt, S., Gilles, E.D.: Minimal cut sets in biochemical reaction networks. Bioinformatics **20**(2), 226–234 (2004)
11. von Kamp, A., Klamt, S.: Enumeration of Smallest Intervention Strategies in Genome-Scale Metabolic Networks. PLoS Comput. Biol. **10**(1), e1003378 (2014)
12. Schuster, S., Hilgetag, C.: On Elementary Flux Modes in Biochemical Reaction Systems At Steady State. J. Biol. Syst. **02**(02), 165–182 (1994)
13. Hädicke, O., Klamt, S.: Computing complex metabolic intervention strategies using constrained minimal cut sets. Metab. Eng. **13**(2), 204–213 (2011)
14. Ballerstein, K., von Kamp, A., Klamt, S., Haus, U.U.: Minimal cut sets in a metabolic network are elementary modes in a dual network. Bioinformatics **28**(3), 381–387 (2012)
15. de Figueiredo, L.F., Podhorski, A., Rubio, A., Kaleta, C., Beasley, J.E., Schuster, S., Planes, F.J.: Computing the shortest elementary flux modes in genome-scale metabolic networks. Bioinformatics **25**(23), 3158–3165 (2009)
16. Rocha, I., Maia, P., Evangelista, P., Vilaça, P., Soares, S., Pinto, J.P., Nielsen, J., Patil, K.R., Ferreira, E.C., Rocha, M.: OptFlux: an open-source software platform for in silico metabolic engineering. BMC Syst. Biol. **4**(1), 45 (2010)
17. Rocha, M., Maia, P., Mendes, R., Pinto, J.P., Ferreira, E.C., Nielsen, J., Patil, K., Rocha, I.: Natural computation meta-heuristics for the in silico optimization of microbial strains. BMC Bioinformatics **9**(1), 499 (2008)
18. Feist, A.M., Henry, C.S., Reed, J.L., Krummenacker, M., Joyce, A.R., Karp, P.D., Broadbelt, L.J., Hatzimanikatis, V., Palsson, B.Ø.: A genome-scale metabolic reconstruction for Escherichia coli K-12 MG1655 that accounts for 1260 ORFs and thermodynamic information. Mol. Syst. Biol. **3**(121), 1–18 (2007)

SCENERY: A Web-Based Application for Network Reconstruction and Visualization of Cytometry Data

Giorgos Athineou, Giorgos Papoutsoglou, Sofia Triantafillou, Ioannis Basdekis, Vincenzo Lagani and Ioannis Tsamardinos

Abstract Cytometry techniques allow to quantify morphological characteristics and protein abundances at a single-cell level. Data collected with these techniques can be used for addressing the fascinating, yet challenging problem of reconstructing the network of protein interactions forming signaling pathways and governing cell biological mechanisms. Network reconstruction is an established and well studied problem in the machine learning and data mining fields, with several algorithms already available. In this paper, we present the first web-oriented application, SCENERY, that allows scientists to rapidly apply state-of-the-art network-reconstruction methods on cytometry data. SCENERY comes with an easy-to-use user interface, a modular architecture, and advanced visualization functions. The functionalities of the application are illustrated on data from a publicly available immunology experiment.

Keywords Cytometry · CyTOF · Network reconstruction · Web application · Signaling pathway

1 Introduction

Signaling networks are well-organized chains of complex molecular events [1]. Molecular stimuli trigger these events by orderly changing the state of specific proteins ultimately perturbing the cell's metabolism, shape, gene expression, or ability to divide. Flow cytometry is a robust and broadly accessible method nowadays, able to provide quantitative measurements on such sensitive macro-molecular

G. Athineou · G. Papoutsoglou · S. Triantafillou · V. Lagani · I. Tsamardinos(✉)
Department of Computer Science, University of Crete, Voutes Campus, 700 13 Heraklion, Greece
e-mail: tsamard@csd.uoc.gr

I. Basdekis
Foundation for Research and Technology, Hellas - Institute of Computer Science,
Vassilika Vouton, 700 13 Heraklion, Greece

M.S. Mohamad et al. (eds.), *10th International Conference on PACBB*,
Advances in Intelligent Systems and Computing 477,
DOI: 10.1007/978-3-319-40126-3_21

interactions [2]. Still, reconstruction of signaling networks from flow cytometry measurements has not become popular, primarily due to the limited molecular quantities the method can measure. Recently, a novel technique called Mass Cytometry was introduced revolutionizing the state of the art [2]. Its inherent ability to investigate more than 30 quantities simultaneously, offers, now, the opportunity to delve deeper into cell signaling networks.

Inducing signaling networks from data can be thought as a Network Reconstruction (NR) problem. NR methods have become increasingly popular in biology, especially for inferring gene-gene interaction networks, with numerous scientific works currently published on this subject [3]. The first successful case of signaling NR in the cytometry field was achieved by Sachs and co-authors [4], followed by several applications [5], [6]. However, NR methods are not yet routinely used on single-cell cytometry data. Arguably, this is mainly due to the intrinsic complexity of the task. Attempting to reconstruct signaling pathways requires knowing in detail the semantics of the data, the peculiarities of the cytometry technology, and all available information on the specific pathway and its components. On the other hand, successfully applying NR algorithms requires mastering all the technicalities of these methods, since inaccuracies in the analysis pipeline are potentially able to invalidate all results [7].

In addition to those intrinsic problems, a horizontal factor that limits broadcasting, sharing and reusing scientific results in this domain is the reluctance towards social media [8] and the limited use of online services that support sharing of credible and accurate results. These tools promote openness, hiding at the same time sensible, core aspects of an experiment of this kind (in order to prevent copyright infringement).

In this work, we present SCENERY (Single CEll NEtwork Reconstruction sYstem), a web-based application specifically devised to allow researchers to apply NR methods on single-cell cytometry data, even with limited knowledge of the technical details of these algorithms. SCENERY interface guides the user through a set of easy steps; from data loading and study design specification, to the set-up of the analysis and results visualization and sharing. Its core is built on R and its modularity grants to easily add extensions, particularly additional NR methods. To the best of our knowledge, SCENERY is the first available software of its kind. Several other applications exist for cytometry data analysis, both as stand-alone software (FlowJo, www.flowjo.com), web-service (CytoBank, [9]) and libraries [10]; however none of these tools provide the user with NR functionalities.

The rest of the paper is structured as follows. First, we provide an overview of SCENERY functionalities and internal architecture. A use case on real, publicly available data is then presented for better illustrating SCENERY capabilities. Future extensions of the application are then discussed in the conclusions.

2 The SCENERY Application

2.1 Software Functionality

SCENERY's users are guided by a wizard through a specific sequence of analysis steps [11], as shown in Fig. 1. The user supplies the data (step 1), defines the computational experiment (step 2) and then sets the execution parameters (step 3). The user may run the analysis, and redefine the execution parameters until the desired outcome is achieved. This sequence of steps also serves as an educational path for less experienced users who are interested in exploring any aspect of the available analysis methods. Analysis output can be exported in various ways, mainly publication-quality figures and standard formats for graph-representation (i.e., Graph Exchange XML Format, GEXF). In future versions of SCENERY the user will be able to share the output of the analysis privately (via email or a repository) or publicly (via social media or blogs accounts) to a group of colleagues for further analysis and discussion. Recent social-media citation practices indicate that scientific content is becoming more and more part of every day conversation, thus increasing chances of citation [12].

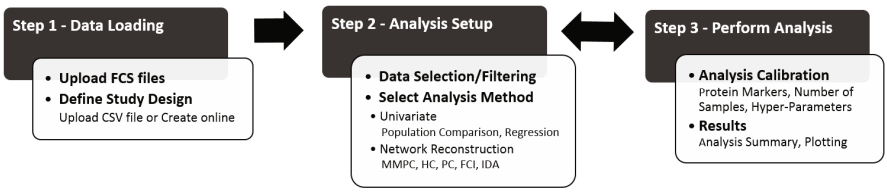
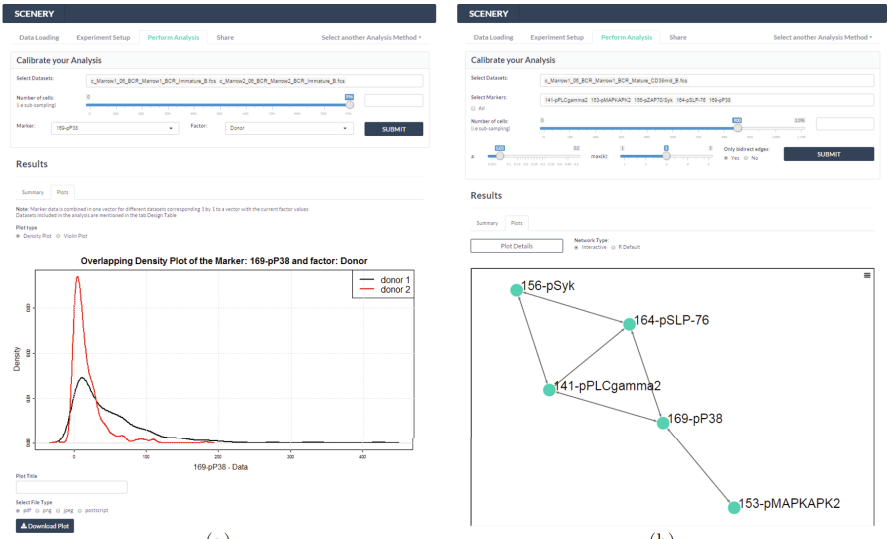


Fig. 1 Flowchart of typical user-application interaction. In step 1 users upload data and define the study design. In step 2 they setup a computational experiment by selecting datasets and the analysis method. In step 3, users calibrate the input parameters and execute the analysis. The analysis can be reconfigured and repeated multiple times.

Data Loading. As first step, the user uploads one or more data files in Flow Cytometry Standard (FCS) format. This format is adopted universally by cytometry analysis software, allowing SCENERY to import and analyze datasets pre-processed by other applications. FCS files may correspond to different samples (e.g. patients, cell types) or conditions (e.g., stimuli, inhibitor dosages). SCENERY goes beyond traditional study design declaration, and it allows users to assert any type of metadata knowledge concerning variables, quantities, attributes, or characteristics of the samples (e.g. gender, age, etc.). Hence, any type and number of factors can be defined in a custom study design, both qualitative (e.g, cell type) and quantitative (e.g., drug dosage). *This flexibility permits to accommodate virtually all possible study designs, both current and future ones*.



(a) (b)

Fig. 2 Visualizing results in SCENERY. (a) Overlapping density plots for the marker p38 on 2 donors. (b) The retrieved reconstructed network after applying MMPC on selected mass cytometry data (see text for details). In both screen-shots the analysis calibration panel is also displayed on top of the graphs, as indicative of the UI.

Analysis Setup. In the second step, the user sets up the desired *computational experiment*. This essentially involves the selection of (a subset of) the uploaded data files on the basis of factors of the study design, and the application of a single data analysis method. Analysis methods included in the current version of SCENERY are subdivided into standard statistical methods and NR algorithms. Statistical analysis methods include the t-test and the analysis of variance for comparisons across different levels of the same factor. Univariate linear and logistic regression methods are also available for modeling the relationship between markers and study design's factors.

SCENERY is the first free software to support a number of NR algorithms for single-cell data. All NR methods represent statistical relationships in the data as networks composed by nodes and edges. Nodes always stand for measurements (e.g., protein abundances). Edges, on the contrary, have different semantics, depending on the type of network the method outputs. Association Networks (AN) connect two nodes with an undirected edge if the corresponding measurements are found statistically associated. Conditional Association Network (CAN) are similar to AN, but associations between nodes are computed conditioning on all (or part) of the remaining measurements. Bayes Networks (BNs) use Directed Acyclic Graphs (DAGs) for representing the multivariate distribution of the data. A common misconception is interpreting a directed edge in a BN as an indication of causal interaction. This is possible only under the standard causal discovery assumptions (Causal Markov Condition, Faithfulness), Even then, not all causal relationships are identifiable by

data alone. Therefore, some algorithms output Partial DAGs (PDAGs), that use directed edges for representing causal relations, and undirected edges to represent edges whose causal direction is unclear. If hidden common confounders are also a possibility, Maximal Ancestral Graphs (MAGs) are typically used instead of BNs. MAGs use directed edges to represent causal relationships, and bi-directed edges to represent confounded relationships. Again, since some causal directions are not identifiable, the algorithms usually output Partial Ancestral Graphs (PAGs) that use circle endpoints to indicate ambiguous orientations.

In this version we included five state-of-the-art NR algorithms. We use the MMPC algorithm [13] for deriving CANs and the HC algorithm [14] for reconstructing BNs. Algorithms [15] PC and FCI are used for reconstructing PAG and MAG models, respectively. Finally, the IDA algorithm [15] allows the estimation of a lower bound for the effect size of the causal relation between two variables.

Perform Analysis. Once an analysis method is selected the wizard redirects to the third step where the user is provided with analysis calibration options. Common options for all methods is deciding which markers and the number of cells to employ for the analysis. Next, the user defines the method-specific hyper-parameters and submits the analysis to the system. The analysis output is presented in a separate results panel. This panel consists of two sections/tabs; namely, Summary and Plots. The Summary tab recapitulates the performed analysis reporting metadata information and a textual overview of the results. In the Plots tab and depending on the selected analysis method a separate graphical (downloadable) representation of the results is included. Regarding the population comparison methods, results are graphically displayed using overlapping density plots (Fig. 2-a) and violin plots while, scatter-plots with fitted regression lines are used for regression analyses. For the NR analysis two network visualization are available (Fig. 2-b), a static and an interactive one. Interactive visualization includes features such as zooming, node re-positioning and several others rendering a user-centric layout.

Finally, data visualization functionalities are available for exploring the data by histograms on stand-alone markers or by scatter-plots for multiple markers.

2.2 Application Architecture

SCENERY is a platform-independent web application of Client-Server architecture; built on R and PHP running on an Apache web server (www.apache.org/). The interface on the Client side is implemented using HTML5, CSS3 for structuring and presenting the content and JavaScript for light-weight tasks such as validation of forms, effects on moving elements, asynchronous communication and more. In order to alleviate the overhead associated with common tasks in web development, the Bootstrap web framework (getbootstrap.com/) is used, while the R Shiny web framework is used (shiny.rstudio.com/) to allow R functions communicate among

Client and Server. All analysis methods are implemented in R and run on the Server. Additionally, PHP was used as an application skeleton/controller, for managing most of the Client-Server interaction and database operations. A MySQL database is used for storing user information and history (www.mysql.com/).

While the current architecture is a work in-progress, future version(s) will be extended with operational information built upon Design Strategy for Device Independence [16] enabling the web application to be utilized in various screen dimensions and environments of use.

Modularity. One of the main features of SCENERY is its modularity. Each analysis method is provided by a single R function with a standardized signature (dataset, method's options) and results' type (summary, visualization). This ensures that further analysis methods can be easily integrated within the step-wised SCENERY structure. We are planning to allow users submitting their own NR methods as R code in future versions of SCENERY.

3 Application on Immunology

3.1 Definition of the Problem

To a computer scientist, signaling networks are informal causal models for which proteins are key members. Their role is to relay the signal by switching between active and inactive states, thereby altering their function. Information is passed on sequentially from one protein to the other until the response is produced. We demonstrate an application of SCENERY by trying to reconstruct a part of a signaling pathway, using public mass cytometry data published in [17]. In the original study, the authors use mass-cytometry to measure 31 proteins related to the human hematopoietic system in two healthy bone marrow donors. Cells were stimulated with several activators to uncover distinct signaling mechanisms. Here, we use data from B-cell populations. Particularly, cells treated with stimulus of the B cell antigen-receptor (BCR). BCR signaling is known to trigger several signaling cascades simultaneously permitting many distinct outcomes [18]. Hence, this dataset provides an excellent showcase for the features and applicability of SCENERY in the research of signaling pathway networks.

3.2 Analyzing Cytometry Data with SCENERY

In the following examples we employ a subset of proteins, known to be involved in BCR signaling, namely SYK, BLNK, PLCγ2, p38 and MAPKAPK2. Fig. 2-a illustrates how SCENERY would visualize a population comparison result (Univariate

Analysis). For this graph we employed data from 2 donors for the protein marker p38. At the upper half, the screen-shot shows the configuration of the calibration options. At the bottom half, the two overlapping density plots for this specific analysis are shown.

In the same spirit, Fig. 2-b displays the NR results as they were retrieved by running the MMPC algorithm in SCENERY. Bi-directed edges denote correlation between the respective protein markers in a sense that both bi-connected nodes have been selected in the Parent-Children set of each other. Starting from the top left corner, then, the reconstructed network indicates that SYK, BLNK and PLCγ2 are inter-correlated. This is true biologically because the stimulated BCR attracts and activates SYK which, in turn, attracts, interacts and phosphorylates both BLNK and PLCγ2 [18],[19]. This process is part of a complex stimulation process that ultimately activates several proteins. One of them is p38 which interacts with both PLCγ2 and BLNK in order to be activated [20]. This process is captured in SCENERYs output and is shown in the reconstructed network by the respective correlation edges. After p38 and further downstream, the reconstructed network extends to MAPKAPK2. This edge is also consistent with the literature, where MAPKAPK2 is found to be directly phosphorylated by p38 [21].

4 Discussion and Conclusions

In this work we introduced SCENERY, the first freely available web-based application for single-cell NR analysis. SCENERY packages advanced machine-learning methods in a user-friendly environment: a wizard guides users through all phases of the complex NR effort delivering a simple-to-use interface. This allows biology researchers unfamiliar with the technical details to exploit NR methods in discovering novel signaling pathways. It also allows SCENERY to serve as an educational tool for exploring the features of NR methods.

We showcase some of SCENERY's features using published mass cytometry B-cell data. We illustrate in Fig. 2-a the simplicity with which the software can represent standard statistical analysis results. In Fig. 2-b we show how users can easily assess results from a NR analysis.

More features will be implemented in feature releases. These include: implementation of an online archive for the users sessions; establish connection with other online services for directly loading public data; standard cytometry pre-processing functionalities (e.g., gating, data compensation); sharing analysis results via email or social media; and enriching SCENERY with even more analysis methods. Particularly, we envision that in the future users will be able to submit their own NR analysis methods as R source code for being added to the functionalities of SCENERY. These custom methods may be re-utilized by colleagues of members of a group specified by the end-user.

Our efforts towards this open-source approach hold the promise to transform SCENERY in an essential tool for the cytometry community for understanding the organization of complex cellular processes such as signaling networks.

Acknowledgments This work was funded by European Research Council (ERC) and is part of the CAUSALPATH - Next Generation Causal Analysis project, No 617393. We sincerely thank Karen Sachs, David Gomez-Cabrero, Angelika Schmidt and Jesper Tegner for their invaluable comments, suggestions and encouragement on the start of this project.

Availability. Instructions on how to access and use SCENERY are available at http://mensxmachina.org/en/software/.

References

1. Ullrich, A., Schlessinger, J.: Signal transduction by receptors with tyrosine kinase activity. Cell **61**(2), 203–212 (1990)
2. Bendall, S., Nolan, G., Roederer, M., Chattopadhyay, P.: A Deep Profilers Guide to Cytometry. Trends Immunol. **33**(7), 323–332 (2012)
3. Marbach, D., Costello, J.C., Kffner, R., et. al.: Wisdom of crowds for robust gene network inference. Nat. Methods **9**(8), 796–804 (2012)
4. Sachs, K., Perez, O., Pe'er, D., Lauffenburger, D.A., Nolan, G.P.: Causal Protein-Signaling Networks Derived from Multiparameter Single-Cell Data. Science **308**, 523–529 (2005)
5. Itani, S., Ohannessian, M., Sachs, K., et. al.: Structure learning in causal cyclic networks. In: JMLR Workshop and Conference Proceedings (2010)
6. Qiu, P., Simonds, E.F., Bendall, S.C., et. al.: Extracting a cellular hierarchy from high-dimensional cytometry data with SPADE. Nat. Biotech. **29**, 886–891 (2011)
7. Lagani, V., Triantafillou, S., Ball, B., Tegner, J., Tsamardinos, I.: Probabilistic Computational Causal Discovery for Systems Biology. A Computational Modeling Approach. In: Uncertainty in Biology (2015)
8. Bik, H.M., Goldstein, M.C.: An Introduction to Social Media for Scientists. PLoS Biol. **11**(4), e1001535 (2013). doi:10.1371/journal.pbio.1001535
9. Kotecha, N., Krutzik, PO., Irish, JM.: Web-based Analysis and Publication of Flow Cytometry Experiments. Curr. Prot. Cyt., Chapter 10, Unit10.17. (2010)
10. Levine, J.H.., Simonds, E.F., Bendall, S.C., et. al.: Data-Driven Phenotypic Dissection of AML Reveals Progenitor-like Cells that Correlate with Prognosis. Cell (2015)
11. Kelkar, S. A.: Usability and Human-Computer Interaction: A Concise Study, p. 306 (2009)
12. Priem, J., Costello, K.L.: How and why scholars cite on Twitter. Proceedings of ASIST **47**(1), 104 (2010)
13. Lagani, V., Athineou, G., Borboudakis, G. and Tsamardinos, I.: MXM: Discovering Multiple, Statistically-Equivalent Signatures. R package version 0.4.3. (2015). http://CRAN.R-project.org/package=MXM
14. Scutari, M.: Learning Bayesian Networks with the bnlearn R Package. JSS **35**(3), 1–22 (2010). http://www.jstatsoft.org/v35/i03/
15. Kalisch, M., Maechler, M., Colombo, D., Maathuis, M.H., Buehlmann, P.: Causal Inference Using Graphical Models with the R Package pcalg. JSS **47**(11), 1–26 (2012). http://www.jstatsoft.org/v47/i11/

16. Karampelas, P., Basdekis, I. and Stephanidis, C.: Web user interface design strategy: Designing for device independence. In: C. Stephanidis (ed.). Proceedings of the 13th International Conference on Human-Computer Interaction HCI International 2009, July 19-24, San Diego, CA, USA., pp. 515–524 (2009)
17. Bendall, S.C., Simonds, E.F., et al.: Single-Cell Mass Cytometry of Differential Immune and Drug Responses Across a Human Hematopoietic Continuum. Science **332**(6030), 687–696 (2011)
18. Dal Porto, J., Gauld, S., Merrell, et.al.: B cell antigen receptor signaling 101. Mol. Imm. **41**, 599–613 (2004)
19. Ishiai, M., Kurosaki, M., Pappu, R., et al.: BLNK Required for Coupling Syk to PLCg2 and Rac1-JNK in B Cells. Imm. **10**, 117–125 (1999)
20. Guinamard, R., Signoret, N., Ishiai, et al.: B cell antigen receptor engagement inhibits stromal cell-derived factor (SDF)-1alpha chemotaxis and promotes protein kinase C (PKC)-induced internalization of CXCR4. J. Exp. Med. **189**(9), 1461–1466 (1999)
21. Cargnello, M., Roux, P.: Activation and Function of the MAPKs and Their Substrates, the MAPK-Activated Protein Kinases. Microbiol. Mol. Biol. Rev. **75**(1), 5083 (2011)

Reconstruction of Metabolic Models for Liver Cancer Cells

Jorge Ferreira, Sara Correia and Miguel Rocha

Abstract The liver is one of the largest organs of the adult body and most of its tissue is formed by hepatocyte cells, the main site of the metabolic conversions underlying its diverse physiological functions. Hepatocellular carcinoma is one of the most important human cancers. Genome-Scale Metabolic Models (GSMMs), mathematical representations of the cell metabolism in different organisms including humans, are useful tools to simulate metabolic phenotypes and understand metabolic diseases. In the last years, a few algorithms have been developed to generate tissue-specific metabolic models that allow the simulation of phenotypes for distinct cell types/tissues. This work based on general template GSMMs, which are integrated with available *omics* data. In this work, we propose to develop a pipeline for the systematic evaluation of these algorithms in the creation of models for regular hepatocytes and cancer cell lines, addressing the comparison of the final models obtained.

Keywords Tissue-specific genome-scale metabolic models · Liver metabolism · Hepatocellular carcinoma

1 Introduction

Metabolism is, in a simplified view, the set of chemical reactions occurring in cells, being the main resource for cell stability and viability, enabling it to endure several disturbances, balancing the production of molecules, energy and cell components. The main molecules responsible for the occurrence of reactions are enzymes, which can be regulated by the cell or external signaling. As expected, a small set of modifications on the normal functioning of metabolism can implicate several serious consequences. Diseases like obesity, diabetes, cancer are some of the results of disturbances of the metabolism [1].

J. Ferreira(✉) · S. Correia · M. Rocha
Centre Biological Engineering, University of Minho, Braga, Portugal
e-mail: jhmdf@hotmail.com

M.S. Mohamad et al. (eds.), *10th International Conference on PACBB*,
Advances in Intelligent Systems and Computing 477,
DOI: 10.1007/978-3-319-40126-3_22

Genome-Scale Metabolic Models (GSMMs), under the more general framework of constraint-based modeling, have been widely used to study cell metabolism. GSMMs contain a network of biochemical reactions of a certain organism, which can be represented in a mathematical format [2]. Over the last years, a few human metabolic models have been developed, such as the Edinburgh Human Metabolic Network [3], Recon 1 [4], Recon 2 [5] and Human Metabolic Reaction 2 (HMR 2) [6]. Also, there are some relevant databases including information on human metabolism data, such as HumanCyc [7] and Reactome [8]. These models, when associated with information on environmental conditions (e.g. growth medium) and *omics* data, can provide a reasonable prediction of several metabolic phenotypes, such as growth rates, nutrient uptake rates, compound excretion rates or gene essentially [9].

The most popular method for phenotype prediction, Flux Balance Analysis (FBA) [10], has been widely accepted to study cell physiology within a constraint-based analysis, assuming a pseudo-steady state, i.e. that the internal metabolites are in equilibrium, and also that cells tend to optimize given objective functions, where the most common is to maximize cell growth.

In the past few years, using GSMMs, scientists have been able to simulate cancer cells' metabolism to design drug targets, study oxidative stress and tumor suppressors [11, 12, 13, 14]. As an example, Folger et al. studied the tricarboxylic acid (TCA) cycle fumarate hydratase (FH) enzyme (with a loss-of-function mutation, common in hereditary leiomyomatosis and renal-cell cancer) and the heme pathway, showing that cancer cells could be a target to a specific drug without damaging "wild-type" cells [11].

However, the human organism is complex and includes a huge number of cell types, each with different metabolic functions. So, it is imperative the development of tissue-specific metabolic models for a better understanding of their diseases and metabolic phenotypes. The establishment of several models that can simulate diverse cell types from human tissues may be a good starting point for a better understanding of complex diseases [13].

The human liver is one of the most important organs in the regulation of the human metabolism, being responsible for numerous functions, as the production of bile, removal of toxic substances, decomposition of red cells and chemical regulation of the plasma [1]. The liver consists in different types of cells: parenchymal cells (hepatocytes and bile duct cells) and nonparenchymal cells. Disorders in the metabolism of distinct cell types cause a number of diseases, like hepatitis, nonalcoholic fatty liver disease or hepatocellular carcinoma (HCC) [15]. HCC strikes about half a million humans in the world and it is the most usual form of primary cancer [16]. The analysis of the differences at a molecular level of healthy and disease states, made possible by the enhanced high throughput technologies and decreasing costs of obtaining different *omics* data, can help to clarify the functional mechanisms of liver cells and related diseases [17].

To have a better understanding of how liver cells work, different algorithms have been applied to reconstruct tissue specific metabolic models for hepatocytes [18, 19]. Also, Gille et al. built a manually curated Genome-Scale Metabolic Network for hepatocytes, the HepatoNet1 [20]. In previous work, the authors have done a systematic

analysis of the behaviour of distinct algorithms for tissue-specific reconstruction, using liver cells as a case study [21].

In this work, the objective is to extend previous work, by considering normal hepatocytes, but also by reconstructing models for a liver cancer cell line (HepG2) [22]. Also, we have extended the set of tested algorithms to consider more recent proposals. The aim is to analyse the obtained models for the normal and cancer cells, comparing their structure and functional capabilities, and checking how the algorithm used for the reconstruction affects the results. We aim for a better understanding on how metabolism can differ between "normal" and cancer cells by highlighting the reactions/pathways that are affected.

2 Materials and Methods

2.1 Models and Data

In this work, we will use the Recon 1 as the GSMM, which accounts for 3742 reactions, 2766 metabolites, 2004 proteins and 1905 genes [4]. Several sources of *omics* data were used in this work. Proteomics data were retrieved from the Human Protein Atlas (HPA) [23], which contains protein information. Here, we used HPA data (version 14) for the HepG2 cell line derived from a hepatocellular carcinoma [22] and hepatocytes from normal liver tissue data. Based on these main inputs, the reaction scores will be calculated using the gene-protein rules (GPRs) present in the Recon 1 model, where the logical value "OR" will be replaced by the maximum and "AND" for the minimum of gene scores obtained by the *omics* data.

2.2 Algorithms for Tissue-Specific Model Reconstruction

There are several approaches to create tissue-specific metabolic models based on a generic human model. Here, we briefly explain the four algorithms that will be used in this work.

INIT/tINIT. The Integrative Network Interface for Tissues (INIT) algorithm tries to maximize the number of matches between reaction states (active or inactive) and data states (expressed or not), returning flux values and a context-specific model (i.e. a set of active reactions from the original model). The method solves a Mixed Integer Linear Program (MILP), where binary variables represent the presence of each reaction from the original model in the final one. INIT uses proteomic evidences from HPA, integrating gene expression data when these are missing. An objective function is built assigning positive weights to reactions with high evidence from the data, and negative ones to reactions with low or no evidence. If there are metabolomics

data that supports the existence of a certain metabolite, the algorithm attempts to activate the reactions needed to produce it [13]. The Task-driven INIT (tINIT) is an extension of the previous method [24]. Here, it is possible to define context specific metabolic tasks that the final model needs to perform. These may represent the consumption or production of a metabolite, or the activation of a pathway for a specific tissue.

MBA. In contrast with the previous, the Model-Build Algorithm (MBA) [18] only returns a model, and not reaction fluxes. This algorithm takes as inputs a generic metabolic model and two sets of reactions. The first set contains reactions with a high likelihood of being present in the specific model (C_H), information that comes from a well-curated source (e.g. literature), while the second set contains reactions of moderate likelihood (C_M), typically derived from high-throughput data. Based on these sets, the algorithm iteratively removes non-core reactions from the original model in a random order, and validates if the model remains consistent. The process finishes when all non-core reactions have been tested for removal. As a result, the algorithm generates a specific model with all reactions from C_H, a maximum number or reactions from C_M and a minimal number of other reactions, needed for connectivity purposes. Because the order of reaction removal can affect the final result, the algorithm should be repeated a large number times, to obtain a population of models. The next step is to rank the frequency of the reactions in those models, and adding them in that order to the C_H core, until a coherent model is obtained [18].

mCADRE. The Metabolic Context specificity Assessed by Deterministic Reaction Evaluation (mCADRE) [19] algorithm is similar to the MBA, but in only requires the creation of a single model. It starts by ranking the reactions according to three different scores: expression, connectivity, and confidence. This helps to establish the core set of reactions (based on a threshold value) and also the order by which the non-core ones are removed. The mCADRE algorithm does not consider the levels of expression, but rather the frequency of expression states in a collection of profiles, requiring a previous transformation of the expression data to binary values. Also, reactions are ranked according to their connectivity to adjacent reactions. Finally, ranking based on confidence levels answers for the evidences supporting that particular reaction in the GSMM. During the reconstruction of the tissue-specific metabolic model, the removal of non-core reactions is tried in the previous order, being the reaction removed if it does not disrupt the production of essential metabolites and the core reactions are preserved. However, the algorithm is more flexible than MBA, since it accepts the elimination of some core reactions, in specific situations.

FastCORE. Like MBA or mCADRE, FastCORE's [25] objective is to produce a model where all core reactions remain unaltered, while using a different algorithmic strategy. FastCORE solves two Linear Problems (LP): the first tries to maximize the number of reactions present in the core by comparing the values of a reaction

with a small positive constant, while the other decreases the number of reactions that are not present in the core by minimizing the L_1-norm of the flux vector. Both LPs are alternatively and repeatedly applied until the core is consistent (all core reactions are activated with the minimum of non-core reactions). In the case of reversible ones, FastCORE evaluates both directions.

3 Results and Discussion

For this particular study, we used HPA data for two distinct conditions (normal hepatocytes and tumor cells from HepG2 cell line) as input for the tissue-specific model reconstruction process. Since we are using the Recon 1 metabolic model as a template model, we filtered the data only considering the genes present in the model. Afterwards, we reconstructed tissue-specific metabolic models for each condition using the four available approaches defined in the previous section. In the case of MBA, we generated 50 candidate models in different runs and assembled them in a single model. A model was created per algorithm and per condition, which results in a total of 8 tissue-specific metabolic models. For each algorithm, different parameters were used to reconstruct tissue specific models. The thresholds for construction of the core both for MBA, FastCORE and mCADRE were *Moderate* or *High* reactions. For the tINIT algorithm, a set of specific metabolic tasks that the cell needs to perform were given.

For a visual understanding of the results, we display in Figure 1 the number of reactions that are shared for each set of conditions, using Venn Diagrams.

Next, we performed a hierarchical clustering of the 8 models, to identify relations between algorithms or conditions. With this procedure, we will try to identify the similarities between the algorithms and conditions to uncover their relations. The results are shown in Figure 2. Looking at the results, we can conclude that there are three major groups: the first contains normal cells from mCADRE, MBA and tINIT; the second contains HepG2 models from the same algorithms; while the last incorporates FastCORE models for both cell types. It was expected that the models would group by condition, which occurred with MBA, tINIT and mCADRE but not with FastCORE. This can be explained by the algorithm behaviour, which generates a core of reactions, that seems to be similar in both conditions.

We also performed an enrichment analysis of the differences in Gene Ontology (GO) terms of the two conditions using the same algorithm, using a p-value threshold of 0.025 (using the *Category* and *GOstats* packages from *Bioconductor*). The aim was to evaluate which biological processes were lost and acquired comparing by the cancer cells as compared to normal ones. The processes lost in MBA models were mostly related to the production of small molecules, like nucleotides, while there were gains in the pathways related to the ability to metabolize fatty acids. Regarding the mCADRE algorithm, the biological processes lost were similar to the MBA, while there are gains in pathways related to ions transport. Regarding the tINIT model, there are losses in functions related to the ability to metabolize fatty acids

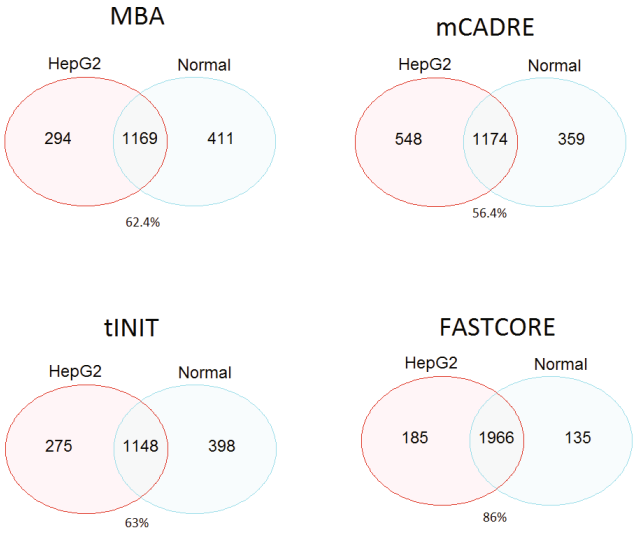


Fig. 1 Differences and similarities in reactions between Normal and Cancer cells from HPA and GEB data using tINIT, mCADRE, MBA and FastCORE algorithms. The values under each Venn diagram represents the percentage of shared reactions for both conditions.

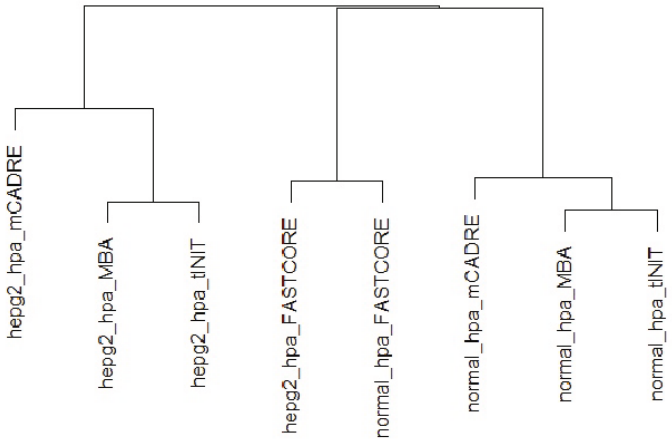


Fig. 2 Hierarchical Clustering of all the 8 models generated with the method "complete".

and gains in biological processes related to the production of ATP and nucleotides. The FastCORE algorithm analysis revealed that the HepG2 model has an increase of the production of ATP (like the tINIT one) and of small molecules through another pathways.

As a final analysis, we decided to evaluate the performance of the models by verifying how many liver-specific metabolic tasks (from [20]) they could complete.

Table 1 Percentage of performed tasks by condition and algorithm of the 281 tasks that Recon 1 is able to perform.

	MBA	mCADRE	tINIT	FastCORE
Normal	7.8%	9.6%	87.9%	58.4%
HepG2	63%	2.5%	92.5%	40.5%

From a total of 442 tasks, Recon 1 can perform 281. The Table 1 illustrates the percentage of tasks that our tissue specific models can perform.

Overall, HepG2 models perform more tasks than the normal hepatocytes ones, with the exception of the models created by mCADRE, the algorithm with the worst performance, and FastCORE. This discrepancy in the other algorithms (mainly in MBA) can be explained due to the fact that there are more genes in the data from HPA that can be mapped to the Recon 1 model in HepG2 (171 genes missing) when compared to regular hepatocytes (360 missing). It was expected that the tINIT algorithm presented the best results due to the fact that the algorithm was developed based on the use of HPA.

4 Conclusions

Liver cancer is still a disease which kills millions of humans per year and although all the biological and technological advances, there are some questions that need a better answer. Although this work is preliminary, there are some considerations to do. Although we use the same data source (HPA) for the reconstruction of tissue-specific models, the algorithms were able to differentiate from normal to HepG2 (with the exception of FastCORE algorithm, which grouped the two different conditions). The next step on the analysis of the generated models were to evaluate the gains and losses of biological processes between conditions. The tINIT algorithm was the one with the more interesting results (loss of the ability to metabolize fatty acids and an increased production of ATP), mainly due to the fact that the algorithm was developed to work with HPA data. The evaluation of the models generated with a set of metabolic functions that are required for the normal function of the hepatocytes. As expected, the tINIT tissue-specific metabolic models (both for normal and HepG2) showed the higher number of tasks performed (around 90%). Although this number is high, we need to take in account that we only used one type of data source and that some tasks could not be made by the Recon 1 model. With this in mind, we need to be able to integrate other types of data sources, try another algorithms and the use of another generic human model to increase the knowledge about the liver metabolism.

References

1. Tortora, G.J., Derrickson, B.H.: Principles of anatomy and physiology. Wiley, Hoboken, New Jersey, USA (2012)
2. Price, N.D., Reed, J.L., Palsson, B.Ø.: Genome-scale models of microbial cells: evaluating the consequences of constraints. Nature Rev. Microbiology **2**(11), 886–897 (2004)
3. Hao, T., Ma, H.-W., Zhao, X.-M., Goryanin, I.: Compartmentalization of the edinburgh human metabolic network. BMC Bioinformatics 11(1) (2010)
4. Duarte, N.C., Becker, S.A., Jamshidi, N., Thiele, I., Mo, M.L., Vo, T.D., Srivas, R., Palsson, B.Ø.: Global reconstruction of the human metabolic network based on genomic and bibliomic data. PNAS **104**(6), 1777–1782 (2007)
5. Thiele, I., Swainston, N., Fleming, R.M., Hoppe, A., Sahoo, S., Aurich, M.K., Haraldsdottir, H., Mo, M.L., Rolfsson, O., Stobbe, M.D., et al.: A community-driven global reconstruction of human metabolism. Nature Biotech. **31**(5), 419–425 (2013)
6. Mardinoglu, A., Agren, R., Kampf, C., Asplund, A., Uhlen, M., Nielsen, J.: Genome-scale metabolic modelling of hepatocytes reveals serine deficiency in patients with non-alcoholic fatty liver disease. Nature Commun. 5 (2014)
7. Romero, P., Wagg, J., Green, M.L., Kaiser, D., Krummenacker, M., Karp, P.D.: Computational prediction of human metabolic pathways from the complete human genome. Genome Biology **6**(1), R2 (2004)
8. Milacic, M., Haw, R., Rothfels, K., Wu, G., Croft, D., Hermjakob, H., D'Eustachio, P., Stein, L.: Annotating cancer variants and anti-cancer therapeutics in reactome. Cancers **4**(4), 1180–1211 (2012)
9. Covert, M.W., Knight, E.M., Reed, J.L., Herrgard, M.J., Palsson, B.O.: Integrating high-throughput and computational data elucidates bacterial networks. Nature **429**(6987), 92–96 (2004)
10. Orth, J.D., Thiele, I., Palsson, B.Ø.: What is flux balance analysis? Nature Biotech. **28**(3), 245–248 (2010)
11. Folger, O., Jerby, L., Frezza, C., Gottlieb, E., Ruppin, E., Shlomi, T.: Predicting selective drug targets in cancer through metabolic networks. Molecular Systems Biology 7(1), (2011)
12. Frezza, C., Zheng, L., Folger, O., Rajagopalan, K.N., MacKenzie, E.D., Jerby, L., Micaroni, M., Chaneton, B., Adam, J., Hedley, A., et al.: Haem oxygenase is synthetically lethal with the tumour suppressor fumarate hydratase. Nature **477**(7363), 225–228 (2011)
13. Agren, R., Bordel, S., Mardinoglu, A., Pornputtapong, N., Nookaew, I., Nielsen, J.: Reconstruction of genome-scale active metabolic networks for 69 human cell types and 16 cancer types using init. PLoS computational biology **8**(5), e1002518 (2012)
14. Jerby, L., Wolf, L., Denkert, C., Stein, G.Y., Hilvo, M., Oresic, M., Geiger, T., Ruppin, E.: Metabolic associations of reduced proliferation and oxidative stress in advanced breast cancer. Cancer Research **72**(22), 5712–5720 (2012)
15. Baffy, G., Brunt, E.M., Caldwell, S.H.: Hepatocellular carcinoma in non-alcoholic fatty liver disease: an emerging menace. Journal of Hepatology **56**(6), 1384–1391 (2012)
16. Jemal, A., Bray, F., Center, M.M., Ferlay, J., Ward, E., Forman, D.: Global cancer statistics. CA **61**(2), 69–90 (2011)
17. Kampf, C., Mardinoglu, A., Fagerberg, L., Hallström, B.M., Edlund, K., Lundberg, E., Pontén, F., Nielsen, J., Uhlen, M.: The human liver-specific proteome defined by transcriptomics and antibody-based profiling. The FASEB Journal **28**(7), 2901–2914 (2014)
18. Jerby, L., Shlomi, T., Ruppin, E.: Computational reconstruction of tissue-specific metabolic models: application to human liver metabolism. Molecular Systems Biology 6(1) (2010)
19. Wang, Y., Eddy, J.A., Price, N.D.: Reconstruction of genome-scale metabolic models for 126 human tissues using mcadre. BMC Systems Biology **6**(1), 153 (2012)
20. Gille, C., Bölling, C., Hoppe, A., Bulik, S., Hoffmann, S., Hübner, K., Karlstädt, A., Ganeshan, R., König, M., Rother, K., et al.: Hepatonet1: a comprehensive metabolic reconstruction of the human hepatocyte for the analysis of liver physiology. Molecular Systems Biology 6(1) (2010)

21. Correia, S., Rocha, M.: A critical evaluation of methods for the reconstruction of tissue-specific models. In: Proc. 17th Portuguese Conference on Artificial Intelligence, EPIA 2015, Coimbra, September 8-11, 2015, pp. 340–352 (2015)
22. Knowles, B.B., Howe, C.C., Aden, D.P.: Human hepatocellular carcinoma cell lines secrete the major plasma proteins and hepatitis b surface antigen. Science **209**(4455), 497–499 (1980)
23. Uhlen, M., Oksvold, P., Fagerberg, L., Lundberg, E., Jonasson, K., Forsberg, M., Zwahlen, M., Kampf, C., Wester, K., Hober, S., et al.: Towards a knowledge-based human protein atlas. Nature Biotech. **28**(12), 1248–1250 (2010)
24. Agren, R., Mardinoglu, A., Asplund, A., Kampf, C., Uhlen, M., Nielsen, J.: Identification of anticancer drugs for hepatocellular carcinoma through personalized genome-scale metabolic modeling. Molecular Systems Biology 10(3) (2014)
25. Vlassis, N., Pacheco, M.P., Sauter, T.: Fast reconstruction of compact context-specific metabolic network models. PLoS Comput. Biol. 10(1) (2014)

Author Index