Analysis and integration of heterogeneous large-scale genomics data

Marine Louarn

UNIVERSITÉ DE RENNES 1

THÈSE DE DOCTORAT DE

L'UNIVERSITE DE RENNES 1

Ecole Doctorale N° 601
Mathématiques et Sciences et Technologies de l'Information et de la Communication
Spécialité : *Informatique*

Par

Marine LOUARN

Analysis and integration of heterogeneous large-scale genomics data

Application to B cell differentiation and Follicular Lymphoma non coding mutations

Thèse présentée et soutenue à RENNES, le 26 Novembre 2020
Unité de recherche : INSERM et IRISA

Rapporteurs avant soutenance :

Adrien COULET	MCU, LORIA Nancy
Lydie LANE	Chercheuse, SIB Lausanne, Suisse

Composition du jury :

Président :	Alexandre TERMIER	Professeur, Université Rennes1 Rennes
Examinateurs :	Salvatore SPICUGLIA	Directeur de recherche, INSERM U1090 Marseille
	Sarah COHEN BOULAKIA	Professeure, LRI Orsay
	Fabrice CHATONNET	Ingénieur de recherche, CHU Rennes
Dir. de thèse :	Anne SIEGEL	Directrice de recherche CNRS, IRISA Rennes
Co-dir. de thèse :	Thierry FEST	PU-PH, INSERM / CHU Rennes

Invité(s)

Olivier DAMERON	Professeur, Université Rennes1 Rennes

“I’m sure there are words that are simply in there ’cause I like them. I know I couldn’t justify each and every one of them.”

Neil GAIMAN

ACKNOWLEDGEMENT

I would like to sincerely thank Lydie Lana and Adrien Coulet for accepting to review this thesis. I would also like to thank Alexandre Termier, Salvatore Spicuglia and Sarah Cohen Boulakia for accepting to be part of my jury. I would also like to thank Anne Siegel, Olivier Dameron, Fabrice Chatonnet and Thierry Fest for offering me that internship - which now feels likes ages ago - that lead to this Ph.D. That you for the mentoring, the support and the push to better myself scientifically. A huge thank to the Dyliss team, but also GenOuest and GenScale, it has been a pleasure working with you guys, laughing together during breaks and thanks for the overall good atmosphere that made those three and a half year a lot more fun. A special thank for Xavier, for his continuous support on AskOmics and for answering my numerous questions. And special thank to the 'Midi les doctorants' group, with all our silly yet fascinating discussions about basically everything, and for some of my best laugh also. They are too many names to give a full list but from the bottom of my heart, thanks you ! You made this thesis a lot more fun. Thanks also for hearing me venting when everything was going wrong. Thanks to the INSERM group for enduring my computer sciences presentations and helping me better understand the biological context of my thesis.

Merci Arty et Céleste pour la *Boirlothe* devenue traditionnelle. Merci a tous les amis qui m'ont aidée à rendre ses années inoubliables, que ce soit sur cette campagne de D&D qui dure depuis aussi longtemps que ma thèse maintenant ou simplement autour d'un verre ou d'un mug. Un merci tout spécial à Pol, comme tu m'as dit « Merci et bravos à tous ». Un immense merci à mes parents, qui ont toujours cru en moi et qui m'ont encouragée au quotidient. Merci de m'avoir soutenue et aidée à faire des études aussi longues (10 ans...). Merci papa pour le service technique. Merci maman pour les réserves de sucres et la couture ! Merci Amaury, ce fut fun de partager ces trois ans avec toi. Certes tu auras sans doute ta thèse avant, mais je garde l'avantage de l'âge et du nombre de diplômes et ça, c'est ce qui importe. Enfin, Mamick, je sais que tu ne peux plus lire ces mots, mais je sais que tu serais fière de moi. Merci pour tout.

1. If Covid allows.

9		7			8			
					3	9		8
8		3		6		5		
	4				6	2		
6					2		1	4
		2	4				6	5
2			6	7	4		8	
		4						
1			2	3	9	4	5	

FIGURE 1 – For the friends and folks who came[1] in support, but may not understand much.

Please find the solution of this sudoku in the Ph.D thesis of Amaury LOUARN, called "A topological approach to virtual cinematography." (*Université de Rennes 1*, November 23rd, 2020).
Here is the solution to the sudoku proposed in his thesis :

2	8	5	6	4	9	1	7	3
3	7	9	8	5	1	6	4	2
6	1	4	2	3	7	8	9	5
1	6	3	7	9	5	2	8	4
9	4	8	1	6	2	3	5	7
7	5	2	4	8	3	9	1	6
4	3	1	5	2	8	7	6	9
5	9	7	3	1	6	4	2	8
8	2	6	9	7	4	5	3	1

TABLE OF CONTENTS

RÉSUMÉ EN FRANÇAIS

Les sciences de la vie et particulièrement la santé sont de gros producteurs de données (150 exabytes en 2010 pour le seul système de santé Américain [COTTLE et al., 2013]). En plus de la quantité de données produites, s'ajoutent les défis de leur diversité et de leur hétérogénéité, puisque ces données sont de natures complémentaires (imagerie, diagnostique clinique, données génomiques, etc.) et proviennent de nombreuses sources.

Nous nous focalisons sur les réseaux de régulation de gènes, un sous-domaine des sciences de la vie. Même dans ce contexte spécifique, les données sont nombreuses et hétérogènes. Pour comprendre la régulation de l'expression génique, il est nécessaire d'intégrer des informations de nombreuses expériences sur l'ensemble du génome. En effet, un réseau de régulation de gènes est la somme des interactions entre les régulateurs, ou entre régulateurs et d'autres entités biologiques, dans une cellule afin de diriger l'expression génique. Un réseau de régulation est généralement représenté comme un graphe dont les sommets sont les entités biologiques (gènes, protéines ou métabolites) et les arcs les interactions entre elles (protéine/protéine, protéine/ADN...).

Nous nous intéressons particulièrement aux réseaux de régulation car ils sont généralement perturbés en cas de cancer. Effectivement, les cancers sont des maladies très diversifiées et hétérogènes résultant d'un contexte génétique spécifique à chaque patient. Une partie de cette hétérogénéité peut être expliquée par les mutations génétiques dans les parties de l'ADN codant pour des gènes. Cependant, ces mutations ne suffisent pas à expliquer l'intégralité de la variabilité. Des régions non-codantes de l'ADN sont aussi connues pour intervenir dans la régulation de l'expression des gènes. Cette régulation peut ainsi être perturbée dans le cas de mutations dans les régions non-codantes [KHURANA et al., 2016]. La littérature montre justement que de telles mutations peuvent être à l'origine de l'apparition de tumeurs ou d'en permettre le maintien et la survie [MANSOUR et al., 2014 ; QUEIRÓS et al., 2016].

Dans le cadre de cette thèse, nous nous intéressons à un cancer particulier :

le lymphome folliculaire. Il s'agit du cancer non-Hodgkinien le plus commun et il est caractérisé par une altération du réseau de régulation des cellules B. Il est considéré incurable, car les cas de rechute ou de résistance au traitement sont fréquents. Il est aussi caractérisé par une grande hétérogénéité chez les patients qui en sont atteints, principalement dans l'épigénétique de sa régulation [KORFI et al., 2017].

Cependant pour comprendre les modifications de la régulation dans le lymphome folliculaire, il nous faut comprendre la différenciation saine des cellules B naïves, des cellules impliquées dans la réponse immunitaire. Pour ce faire nous avons accès à des données génomiques (expression génique obtenue par RNA-seq et accessibilité de la chromatine identifiée par ATAC-seq) obtenues sur un nombre restreint d'échantillons correspondant à des populations cellulaires biologiquement proches. Cette différenciation est déjà étudiée et il ressort plusieurs régulateurs connus : PRDM1, BACH2, BCL6, PAX5 et IRF4 [WILLIS et NUTT, 2019]. Cependant, ils ne suffisent pas à expliquer complètement le processus de différenciation. On soupçonne donc que d'autres régulateurs encore inconnus pourraient être importants pour certaines étapes de ce processus de différenciation.

Pour inférer les réseaux de régulations des deux contextes de différenciation des cellules B et de lymphome folliculaire, nous nous appuyons sur les méthodes existantes, en particulier *Regulatory Circuits* [MARBACH et al., 2016], une méthode d'inférence de réseaux de régulations orientée vers les cellules humaines. Cependant comme nous le détaillerons, ces méthodes produisent des réseaux non signés et elles nécessitent une grande quantité de données, ce que nous ne possédons pas. Elles sont aussi fréquemment difficilement réutilisables car simplement décrites en tant qu'algorithmes sans implémentations.

Analyse de *Regulatory Circuits*, une méthode d'inférence de réseaux de régulation

Dans le premier chapitre de cette thèse, nous nous intéressons au projet *Regulatory Circuits*, une des méthodes d'inférence de réseaux de régulation les plus récentes et la plus complète. Elle a pour but d'identifier les réseaux de régulations spécifiques à certains types cellulaires. Le but sous-jacent de cette méthode est de trouver les perturbations régulatrices spécifiques à certaines maladies, but que nous rejoignons

dans le cadre du lymphome folliculaire. Notre première intention a été d'appliquer la méthode de *Regulatory Circuits* sur un nouveau jeu de données, de manière à inférer de nouveaux réseaux de régulation. Cependant, nous avons été confrontés à des limitations méthodologiques et techniques. Nous n'avons pas été en mesure d'utiliser les fichiers résultant de *Regulatory Circuits*, car ils agrègent les quatre populations cellulaires saines auxquelles nous nous intéressons en une seule (Cellules B CD19+) et ne possèdent pas de type cellulaire proche du lymphome folliculaire. Nous avons aussi découvert que l'implémentation fournie ne fonctionne pas et que la section Matériels et Méthodes de l'article ne permet pas de reproduire les résultats intermédiaires fournis par *Regulatory Circuits*.

Nous avons donc proposé deux méthodes permettant de recalculer *Regulatory Circuits* : l'une à partir de l'ensemble fichiers proposés (fichiers d'entrée et fichiers intermédiaires) puis déroulant la méthode décrite dans les méthodes du papier et une seconde utilisant uniquement les fichiers d'entrée et déroulant la même méthode. Cependant, même en reproduisant pas à pas les étapes de la méthode décrite dans l'article nous avons été incapables d'obtenir les même résultats. Ces différents problèmes illustrent les principaux obstacles à la reproductibilité et à la réutilisabilité des données en science de la vie : des jeux de données spécifiques, non réutilisables et dont la méthode d'analyse ou de traitement ne peut pas être reproduite. Notre première contribution a ainsi consisté à réaliser un recensement de ces différentes limitations de *Regulatory Circuits* et de montrer qu'elles peuvent être organisées en catégories généralisables aux autres méthodes.

Ceci nous a conduits à la fois à proposer une méthode de représentation des données permettant de dépasser les problèmes de reproductibilité et de réutilisabilité, et à proposer une nouvelle méthode d'inférence de réseaux de régulations permettant d'analyser des petits jeux de données issus de populations cellulaires biologiquement proches et de distinguer les relations d'activation et d'inhibition.

Les technologies du Web Sémantique comme cadre de la formalisation des méthodes d'inférence de réseaux

Notre seconde contribution a été de proposer une approche basée sur les technologies du Web Sémantique pour implémenter la méthode publiée par *Regulatory Cir-*

cuits de manière à la rentre plus facilement réutilisable et disponible. La méthode de *Regulatory Circuits* est basée sur deux niveaux : un premier à l'échelle des échantillons – basé sur les expériences biologiques – et un second à l'échelle du tissu cellulaire – composé d'au moins un échantillon, mais généralement de plusieurs. Nous montrons que le premier niveau de l'analyse de *Regulatory Circuits* – réseaux spécifiques des échantillons – peut être formalisé en deux requêtes SPARQL avec des performances acceptables : moins de 4 heures pour calculer l'ensemble des réseaux.

Notre approche consiste à formaliser les données et les résultats biologiques en tant que graphe RDF. Nous produisons un modèle de *Regulatory Circuits* qui permet un accès unifié aux réseaux spécifiques des échantillons. Une fois les fichiers d'entrée de *Regulatory Circuits* traités et formatés pour être intégrés, le pipeline peut être assimilé à deux requêtes, cependant il est toujours nécessaire d'effectuer un post-traitement pour pouvoir obtenir les réseaux spécifiques des tissus.

Nous avons ainsi démontré que les technologies du Web Sémantique sont un bon cadre pour la formalisation des méthodes d'inférence de réseaux de régulation en tant que graphe de données, et que cette solution permet d'améliorer la réutilisation de la méthodologie mais aussi son interopérabilité avec des données extérieures. Cependant, à ce stade, nous ne proposons pas de conservation des graphes calculés. De plus cette méthode s'avère plus complexe quand on tente de la passer à l'échelle sur les réseaux tissu-spécifiques.

Utilisation de graphes RDF multi-niveaux pour la structuration de *Regulatory Circuits*

Notre troisième contribution étend le travail de la partie précédente sur *Regulatory Circuits*, de manière à passer à l'échelle les graphes de données RDF et de ne plus seulement inférer les réseaux de régulation à l'échelle des échantillons mais aussi à celle des tissus.

Nous avons élaboré une stratégie permettant de générer une ressource publique contenant à la fois les données biologiques de *Regulatory Circuits*, des données liées à des ressources extérieures suivant le LOD, mais aussi les résultats de l'analyse d'inférence de réseaux de régulations à l'échelle de l'échantillon et de celle du tissu. Pour le graphe de base, représentant les données biologiques, nous utilisons celui développé

dans la section précédente ainsi que les deux requêtes SPARQL l'accompagnant. Les réseaux spécifiques des échantillons alors calculés sont réinjectés dans la base de donnée (endpoint) sous forme de nouveaux graphes RDF nommés. Ces nouveaux graphes sont alors eux-mêmes requêtés pour obtenir la dernière étape du pipeline de *Regulatory Circuits* : les réseaux tissu-spécifiques. Le résultat de ses réseaux est lui-même réinjecté en tant que graphe nommé dans la base de données. Ceci permet de plus facilement requêter une sous-partie des données en explicitant le nom du graphe.

Cette partie s'est avérée plus difficile que prévu, nécessitant des calculs de quasiment un mois entre le calcul des requêtes de manière à obtenir les relations pondérées pour l'ensemble des réseaux – échantillons et tissus - et la ré-injection des graphes nommés au sein de la base de données.

Une nouvelle méthode d'inférence de réseaux

Notre quatrième contribution est une nouvelle méthode d'inférence de réseaux se focalisant sur des petits jeux de données biologiquement proches. Cette méthode permet aussi l'inférence de réseaux signés (activation ou inhibition) contrairement à de nombreuses méthodes existantes. Comme mentionné précédemment, le besoin de développer un nouveau pipeline spécifique à nos données est lié à l'impossibilité de trouver des méthodes proposées avec le niveau de détail représentant nos populations cellulaires et non un unique tissu les recouvrant ainsi que l'inexistence de réseaux de régulation relatifs au lymphome folliculaire. Cela a été renforcé par l'incapacité à reproduire les résultats et la méthode de *Regulatory Circuits* de manière probante.

Cette nouvelle stratégie tire avantage de l'hétérogénéité des données en science de la vie disponibles : information sur l'expression des gènes, sur l'accessibilité de la chromatine, l'activité des régulateurs et la localisation de leurs sites de fixations sur l'ADN. Nous utilisons une normalisation pluri-niveaux des activités appelée "profils" (ou *patterns*), regroupant les gènes et régions d'expression similaire et suivant la même trajectoire à travers les différentes populations cellulaires étudiées. Comme nous nous focalisons sur des petits jeux de données, cette analyse ne peut être statistique et est descriptive. Comme montré précédemment, les technologies du Web Sémantique fournissent un cadre adapté à la création de pipelines d'inférence de réseaux de régulation, c'est pourquoi nous les utilisons pour la création d'un graphe de données en RDF, représentant nos entitées et leurs relations, que nous pouvons ensuite re-

quêter pour obtenir l'ensemble des relations entre régulateurs et gènes. Le reste de cette méthode utilise une logique de raisonnement basée sur la connaissance des experts du domaine pour décrire les relations de régulations potentielles en accord avec le contexte et les règles biologiques. Le pipeline est fait de manière à être volontairement strict, car le but de notre méthode est de proposer une liste des régulateurs potentiels les plus susceptibles d'agir sur les réseaux et non la liste complète des régulateurs. Notre méthode permet d'obtenir des réseaux signés, basés sur la cohérence des activités respectives des gènes, des régions et des régulateurs sur l'ensemble des populations considérées.

Nous avons testé notre pipeline sur des données issues de *Regulatory Circuits* et avons été en mesure d'obtenir des réseaux de régulation cohérents avec leurs données mais avec un meilleur taux de récupération des gènes les moins exprimés, probablement grâce à l'introduction des relations d'inhibition. Nous avons pu aussi vérifier la véracité des signes inférés à l'aide des deux bases de données majeures (Trrust et Signor) avec lesquelles nous sommes en concordance pour 70% des relations inférées existant à la fois dans ces bases et dans nos réseaux.

Application aux cellules B

Notre dernière contribution a été l'application du nouveau pipeline décrit dans la section précédente à un jeu de données de la différenciation des cellules B, de manière à mieux comprendre ce processus biologique et à identifier de nouveaux régulateurs potentiels. Ce jeu contient des données relatives à quatre populations cellulaires biologiquement proches : les cellules B naïves, les cellules B mémoires IgG et IgM ainsi que les plasmablastes. Nous avons extrait 314.965 relations entre régulateurs et gènes, résultant en deux niveaux de réseaux de régulations : un réseau global représentant les relations entre les différents profils d'expressions et plusieurs graphes régulateur-gène spécifiques à chacun de ces profils. Cependant, les graphes produits sont extrêmement denses, nous proposons donc une méthode de post-traitement afin de réduire le nombre de régulateurs pouvant être clefs. Le but est de produire une liste de régulateurs priorisés de manière à pouvoir être testés biologiquement pour prouver leur impact sur les réseaux et valider leur importance inférée. Nous avons défini deux critères : la couverture (la capacité du régulateur à contrôler un grand nombre de cibles) et la spécificité (la capacité du régulateur de ne réguler qu'un seul profil ou une

direction de différenciation).

Nous avons produit une liste contenant 146 régulateurs qui valident ces deux critères et qui sont des candidats régulateurs-clefs de la différenciation. Cette liste inclut notamment : BACH2, PRDM1, PAX5 et IRF4, quatre régulateurs dont nous connaissons l'implication dans la différenciation des cellules B grâce à la littérature. Le seul régulateur présent dans la littérature que nous ne trouvons pas dans nos réseaux est BCL6. Cependant il n'est pas présent dans nos données initiales en tant que régulateur, car nous n'avons pas d'information sur ses sites de fixation. Nous ne l'avons pas non plus détecté dans nos population lors de l'analyse de l'expression des gènes. De fait, BCL6 ne pouvait donc pas apparaître dans les réseaux finaux. Il serait interessant de voir s'il ressort avec l'analyse des données relatives au lymphome folliculaire ou en ajoutant de nouvelles populations qui expriment ce facteur.

Les régulateurs ainsi identifiés auront besoin d'être confirmés à l'aide d'expérimentations biologiques. Notre pipeline a permis de réduire l'espace de recherche des régulateurs de la différenciation des cellules B.

Conclusion

En partant des méthodes existantes d'inférence de réseaux de régulation, nous montrons qu'elles étaient souvent peu reproductibles ou réutilisables. En utilisant le cadre des technologies du Web Sémantique, nous proposons une transformation de ces méthodes sous forme de requêtes SPARQL sur des graphes RDF orientés et nommés. Cette approche augmente la reproductibilité, la disponibilité, la réutilisabilité et permet l'enrichissement avec des bases de données publiques. En nous basant sur cette première contribution, nous avons développé une méthode d'inférence de réseaux signés, adaptée à des jeux de données peu nombreux et biologiquement proches. Notre méthode intègre la connaissance experte et est couplée à des post-traitements pour affiner et réduire l'espace de recherche des régulateurs potentiels. L'application de cette nouvelle méthode à la différenciation des cellules B a permis de retrouver des régulateurs connus de la littérature, et d'identifier de nouveaux candidats. Les perspectives de ce travail sont de mieux prendre en compte les phénomènes biologiques de combinatoire des régulateurs, d'optimiser les méthodes de réduction de l'espace de recherche et d'appliquer cette nouvelle méthode à l'interprétation des mutations des régions régulatrices dans le contexte de cancers incurables comme le

lymphome folliculaire.

CHAPITRE 1

INTRODUCTION

Better understanding the mechanisms driving cancers is a key challenge for systems biology. Cancers are characterized by abnormalities in gene expressions originating from mutations on the DNA. They are also a highly diverse group, with high variability between two patients even of the same disease. In fact each patient presents a different genetic context.

One key tool in the understanding of cancer diversity is gene regulatory networks inference methods, as they allow for a deeper knowledge of the regulatory events taking place. A regulatory network is the sum for all the interactions between either regulators or between regulators and other entities in a cell to orientate the fate direction taken by the cell. A gene regulatory network - or so-called transcriptional regulatory network - describes the interactions between regulators called transcription factors (TF) and their target genes which expression they control - either positively (induction) or negatively (inhibition) - by binding at specific sites on DNA in defined regulatory regions. Perturbations in regulatory networks can be caused either by mutations in the coding part of the DNA - leading to an absence or mis-transcription of TF and/or target genes - or by a mutation in non-coding parts of the DNA. A non-coding mutation can occur at a binding site of a transcription factor and stop it from regulating the transcription of its target genes, or even create a new TF binding site and add new edges to the regulatory network.

This thesis focuses on perturbations of the regulatory networks in follicular lymphoma (FL), a type of hematological cancer. This cancer is known for its high heterogeneity in patients and is considered incurable as most patients relapse or resist treatment. Some of the perturbations of the regulation are already known but they do not explain all the variability of the follicular lymphoma. The follicular lymphoma is an alteration of terminal B cells differentiation. In order to study its regulatory networks, we need to better understand the normal differentiation in a first place to have a ground for

comparison.

In this thesis, we propose a new approach to regulatory network inference based on knowledge graphs. We are able to represent the different entities composing a regulatory network into a graph and to model their interactions. Using the Web Semantic technologies, we can represent each layer of information in a RDF graph, this makes the data easier to query to obtain new information or to extract the relations between several entities. This proved to be an efficient regulatory networks inference method by following the relations between the entities of the regulatory chain.

The overarching goal of this thesis is to propose a regulatory context for the follicular lymphoma and subsequently for B cell differentiation. We are aiming to identify some of the key regulators of these both biological processes.

To answer the goal of this thesis, we first looked into existing network-inference methods and workflows, tried to reproduce them and examined their reproductibility, using the example of *Regulatory Circuits*. Secondly, we structured this existing regulatory-network database using Semantic Web technologies, only representing a first output layer in this step (sample-specific networks). Thirdly, we scaled the RDF structure onto the second layer of *Regulatory Circuits* (tissue-specific networks), which meant adding named graphs to better define the structure. Fourthly, we developed an alternative network inference method, specialised into analyzing sparse and closely-related cell-types and producing signed networks. Finally, we applied this new method to biological data, focusing on B cells differentiation. With this new regulatory network inference method, we provide a list of potential regulators (and regulatory relations) and it will be necessary to biologically validate these regulators to check their real impact on the networks.

CHAPITRE 2

STATE OF THE ART AND BIOLOGICAL CONTEXT

In this chapter we present an overview of the literature applied to our problematic.

We focus on the heterogeneity of the data in life science and health in SECTION 2.1 especially while looking into gene regulatory networks, and their potential perturbations in cancer. We also present the two biological contexts we are invested in: the B cell differentiation and one of its pathological counterparts, the follicular lymphoma. In the following SECTION 2.2 we look at the methods of regulatory networks inference, which type of data they take as input, how they process them and what kind of resulting regulatory networks they produce. Finally, in SECTION 2.3 we look into the ways of storing life science data and the existing life science databases. We present the Semantic Web technologies as a framework for data-structuring and querying. We then introduce AskOmics, a dedicated tool for integration and querying based on Semantic Web technologies that hides the technical graph building and query building from the user in favor of a graphic interface.

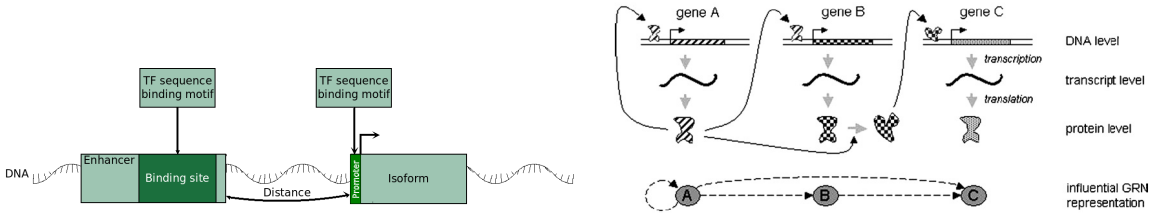
2.1 Heterogeneity of data in Life science

Life science, and in particular health, is a major data producer: for example the US healthcare system as reached 150 exabytes in 2010 [COTTLE et al., 2013], and this trend is expected to increase over the next decade [Z. D. STEPHENS et al., 2015]. In addition to the data quantity challenge data heterogeneity is a second challenge. In [ANDREU-PEREZ et al., 2015] the authors performed a review of different fields of data production in health: from genomics, proteomics, metabolomics, to imaging, clinical diagnosis, patient history and the recent addition of personal devices information

(smart watch, sleep tracker...). Moreover, we are confronted to the rise of multi-omics solutions (Genomics, Epigenomics, Proteomics, Transcriptomics, Metabolomics, etc...) in health and disease related research [KARCZEWSKI et SNYDER, 2018] [HASIN et al., 2017]. Each dataset's size often requires to split the data into several files, which adds another layer of integration and makes global analysis all the more complicated [GOBLE et STEVENS, 2008].

Genes regulatory network We focus on gene regulatory networks, and even in this specific context the data are large and heterogeneous. To understand the regulation in a given context, one needs to perform diverse types of experiment spanning whole genome, currently made available by the recent advent of high throughput sequencing.

A gene regulation network is the sum of the interactions either between regulators, or between a regulator and other entities in a cell to direct the gene expression. A regulatory network is typically represented as a graph composed of nodes representing the genes, proteins or metabolites and of edges representing the interactions (protein/protein, protein/DNA, etc...).



(a) Regulations entities, focusing on the DNA sequence

(b) Regulation, with the addition of the proteins. Figure from [HECKER et al., 2009]

FIGURE 2.1 – Regulation mechanisms

For gene regulatory networks, regulators are specialized proteins called transcription factors (TF) which interact with DNA, the molecular support of genetic information. At the DNA level, a TF will bind to a definite sequence (called a binding motif or binding site) in a specific regulatory region, which should be in an opened 3D conformation to allow the regulation [NARLIKAR et al., 2002] (Fig. 2.1a), and which can be located close (0 kb) or far (500 kb) from its target gene [SMALLWOOD et REN, 2013]. This binding event will then initiate a cascade of molecular events eventually leading to regulation (induction or inhibition) of the target gene expression. This gene can produce

a protein that is itself a TF (see FIGURE 2.1B) and regulates other genes. To act as TF, this protein may need to change conformation or to form a complex with another protein, adding even more complexity to the system.

In biology, the EM algorithm [LAWRENCE et REILLY, 1990] is used to identify and characterize binding sites. This algorithm is limited by the necessity to have at least one occurrence of the binding site in each sequence tested. Most methods currently in use for TF binding sites finding are based on statistical methods [SINHA et TOMPA, 2000], for example the Gibbs Motif Sampler [THOMPSON et al., 2003] based on Markov Chain Monte Carlo use for multiples TF binding sites or YMF [SINHA et TOMPA, 2003] enumerating TF binding site and qualifying them with a z-score. Other methods are based on biological experimentation, using ChIP-seq data (chromatin immunoprecipitation combined with DNA sequencing) of known or suspected TF, to find their binding sites and determine their target by looking at the close potential targets [REN et al., 2000]. In a third category of methods, the gene expression is quantified by RNA-seq with the presence of the TF and without. These methods are mostly used to confirm TF identified with the statistical method such as for the ENCODE consortium [GERSTEIN et al., 2012].

The biological reality is that gene expression is often driven by the combination of several regulators and not unique entities, either with close or distant regulators with synergistic or antagonistic effects. For example, some TFs can work as pairs which change their regulatory impact on the gene's transcription [JOLMA et al., 2015]. Gene expression can also be the output of several concomitant regulatory effects dictated by different TFs. Finally, TFs are themselves regulated by other TFs, which can lead to a cascade of regulations.

Genes regulatory network perturbations in cancer Cancers are heterogeneous diseases for which each patient shows a unique genetic context. Part of this heterogeneity can be explained by mutation of protein-coding genes, but the majority of this heterogeneity is due to mutations outside of the coding regions of the DNA. This non-coding DNA supports the regulatory function of the genes' expression. Mutations in the non-coding areas of the DNA represent 10 times the ones in coding regions [KHURANA et al., 2016]. The impact of those mutations on the cancer development, their resistance to treatment and their likelihood of relapse or transformation is highly unknown,

and is currently understudied. Several studies showed that alteration of the regulatory regions can explain the apparition or the upkeep of tumorous process, either genetic (ex: acute lymphoblastic leukemia [MANSOUR et al., 2014]) or epigenetic (ex: mantle cell lymphoma [QUEIRÓS et al., 2016]). One of the main impacts of non-coding mutations could be through the breaking or the generation of transcription factor binding sites. Thus underlines the importance of regulatory network inference, which can help to prioritize and annotate such non-coding mutations.

Focus on Follicular Lymphoma Follicular Lymphoma (FL) is the most frequent of non-Hodgkin cancer (20% of all cases) and is characterised by the alteration of the regulatory networks of B cells [CARBONE et al., 2019]. In its first steps it is largely asymptomatic leading to late diagnostic, with 70% of the patients presenting stage III or IV. Relapses or treatment resisting cases are frequent, making follicular lymphoma considered incurable.

Follicular lymphoma is characterized by a high heterogeneity in patients, in particular around the epigenetics of the regulation. 85% to 90% of FL are characterized by a t(14;18) translocation placing BCL2 (an anti-apoptotic proto-oncogene) under the control of IGH locus (coding for the heavy chain of B cells antigen receptor), leading to BCL2 over-expression. 50% of FL cases also present genomic alterations impacting the retinoblasma pathway [ORICCHIO et al., 2014]. Other known mutations affect the transcriptional regulation of BCL6 (10-15% of FL cases) [AKASAKA et al., 2003], STAT6 ($>$10%) [YILDIZ et al., 2015], BCL2 [CORREIA et al., 2015], CREBBP and EZH2 [DESMOTS et al., 2019]. Some other works have been done on the modification of the methylation of the DNA [DOMINGUEZ et al., 2017] or the mutation of epigenetic enzymes [JIANG et al., 2016].

However this is not sufficient to explain the diversity of FL and its capacities of relapse or transformation. The study of non-coding mutations in FL could lead to the discovery of new key regulators that explain its variability.

B cells differentiation FL arises during a specific step of the differentiation of Naive B cells (NBC) into antibody producer cells. To be able to understand the potential alterations of gene regulatory networks caused by mutations in non-coding parts of the DNA in FL, we must first identify these networks during the normal differentiation pro-

cess.

As shown in FIGURE 2.2 NBC are mature cells, differentiated in the bone marrow. Their task is to detect any pathogen in the lymph and blood. After an interaction with a pathogen, they migrate into secondary lymphatic organs such as lymph nodes or the spleen. To start their differentiation, NBC interact with other cell types such as T lymphocytes. This differentiation is based on two different mechanisms leading to different Plasma cells (PB). The first mechanism is the extra-follicular differentiation. It is a quick process with low interaction with T lymphocytes and produces plasma cells with a short lifespan, little specificity and affinity but with a really quick response. The second mechanism takes place in the germinal center and takes more time but produces more specific antibodies and with greater affinity to the antigen. During the germinal center reaction, mutations are introduced in the antibody genes of B cells to modify their affinity. This process also produces off-target mutations, which are hypothesized to be at the origin of FL.

After differentiating, the produced plasma cells - which have a very long life span - go to the bone marrow where they produce high levels of antibodies. This differentiation also produces Memory B cells (MBC) with lower affinity but a quick differentiation potential into plasma cells if they encounter the same pathogen again. The immunoglobulin G (IgG) MBC differentiate directly into PB but the immunoglobulin M (IgM) MBC needs to enter the germinal center again [SEIFERT et AL., 2015]. IgG and IgM memory B cells are intermediate between NBC and PB but not in a defined temporal manner. NBC differentiation is coupled to DNA hypomethylation and a specific transcriptional regulation program [BARWICK et AL., 2016]. Looking for the genetic and epigenetic mechanisms behind the faster answer of the memory B cells is also a main issue in immunology. To answer this question, we can identify the involved regulatory networks by looking to gene expression, chromatin accessibility and the reconfiguration of the DNA methylome [CARON et AL., 2015].

Some of the regulators of this differentiation are already known, such as PRDM1, BACH2, BCL6, PAX5 and IRF4 [WILLIS et NUTT, 2019]. PRDM1 and IRF4 are considered to be the master enhancers of NBC to PB differentiation, whereas BACH2, in the contrary, is one of its major repressors [HIPP et al., 2017]. While five known regulators is a good number and more than for some cell types, we know that only those regulators can not explain all the differentiation process and that unknown regulators may remain. For example *Escherichia Coli* regulatory network is formed of 271 TF[MADAN

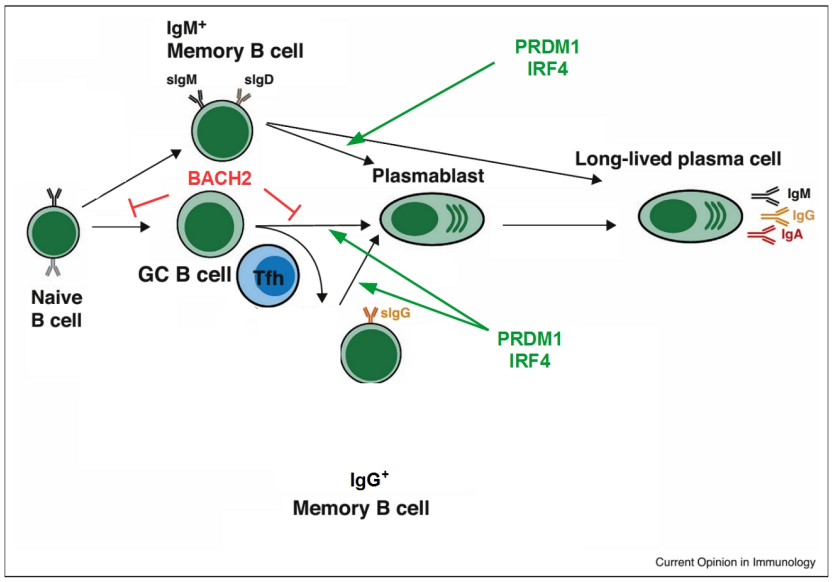


FIGURE 2.2 – NBC differentiation (simplification): Pathways for the generation of human B-cell subsets. Naive B cells can yield IgM-only or Ig class switched memory B cells in a GC dependent manner, as well as plasma cells. MBC can also differentiate into PB, although through distinct mechanisms. [PHAN et TANGYE, 2017]

BABU et TEICHMANN, 2003]. We also do not have a full view on the targets of those regulators and the fine processes of their regulation. We need a better description of their mode of action: do they require another regulator to work as a pair or complex, or is there a cascade of regulatory events?

Conclusion There is a need to better understand regulatory regulations, particularly in cancer where it could explain some of the diversity. In our case we focus on Follicular Lymphoma a specific kind of cancer known for being incurable and with a huge heterogeneity not explained to this day. Some of this diversity could be explained through non-coding mutations, but to be able to interpret them we need to look at the follicular lymphoma regulatory network in comparison to a non-pathological regulation.

2.2 Methods of Regulatory network inference

As described in the previous section, gene expression regulation (also called *transcriptional regulation*) is a major field of investigation in Life science. It allows a better

understanding of major processes such as cell differentiation (how to obtain one or several effective cell types from a common progenitor cell), cell identity (how gene expression is used to define a specific cell type) and cell transformation (how altered gene expression can lead to cell death or cancer) [GARNIS et al., 2004].

Researchers in life sciences and in bioinformatics use huge amounts of data to build extensive regulatory networks from these different entities (genes, TF, regulatory regions), mainly by statistical and machine learning methods. In [HECKER et al., 2009], the authors review four main types of gene regulatory network architectures: information theory models, Boolean networks, differential and difference equations and Bayesian networks. Others workflow and algorithms exist.

In TABLE 2.1, we review the main existing methods of regulatory network inference, according to several different points and the data they use to compute their networks.

Specific entry data and number of data-sets Many of these methods use time series of gene expressions as their only input data (REVEAL, RelNet, BANJO, NIR, ARCANE, TSNI, Mix-CLR, TIGRESS, COALESCE, iRafNet, SINCERITIES, PoLoBa), and for ten methods out these eleven, this is the only mandatory entry while other information can be added but are optional. Unfortunately this require a large set of data on several time points, this can be an issue to produce for disease related regulatory network, in particular for patient specific networks. For example, SINCERITIES was run on 100 cells and 8 time points, TSNI was run using 5 to 10 time points, TIGRESS used 907 experiments and COALESCE on 2200 expression conditions. Unfortunately, with patient experiments we have smaller data sets and less time points.

Samples data: unification of the networks Many of inference methods have been tested in their respective paper on *Escherichia Coli* expression data (NIR, TSNI, COALESCE, DISTILLER, Mix-CLR, TIGRESS, SINCERITIES, PoLoBa). *E. Coli* networks are smaller than human regulatory networks, and often the focus of the regulatory inference was limited to small subset of genes (90 genes for SINCERITIES, maximum of 1,419 for PoLoBag and 100 for Mix-CLR). When not run on *E. Coli* data, many methods are run on in-silico models extracted from the DREAM challenge [1] (PoLoBag:

1. http://dreamchallenges.org/

TABLE 2.1 – Review of different network inference methods

Method name	Reference	Data				Data Normalization			Graph		Type of implementation	Other
		Genes	Regions	TF	Other	2 level discretization	multi-level discretization	continuous	Scored	Signed		
REVEAL	[LIANG et al., 1998]	a(t)				x					Algo	
RelNet	[BUTTE et KOHANE, 1999]	a(t)				x			x		Algo	
BANJO	[HARTEMINK et al., 2000]	a(t)					x				Algo	
NIR	[GARDNER et al., 2003]	a(t)					x				Algo	
ARCANE	[MARGOLIN et al., 2006]	a(t)				x			x		Algo	
TSNI	[BANSAL et al., 2006]	a(t)					x		x		Algo	Focus on 1 gene
COALESCE	[HUTTENHOWER et al., 2009]	a(t)		BS*	nucleosome positioning*, evolutionary conservation*		x		x		C++ implementation & web interface	
DISTILLER	[LEMMENS et al., 2009]	a		BS				x	x		integration: itself mining	Co-expressed genes
Mix-CLR	[MADAR et al., 2010]	a(t)					x		x		Algo	
TIGRESS	[HAURY et al., 2012]	a(t)					x		x		Matlab implementation	
iRafNet	[PETRALIA et al., 2015]	a(t)*, a* Knok-down*		BS*	interaction protein-protein*		x		x		R implementation	
Regulatory Circuits	[MARBACH et al., 2016]		a	BS			x		x		Workflow	
SINCERITIES	[PAPILI GAO et al., 2018]	a(t)						x	x	x	Algo	
PoLoBag	[ROY et al., 2020]	a(t)						x	x	x	Algo	

a = activity, a(t) time series of the activity, * optional, BS = TF binding site, Algo = description of the algorithm without implementation

DREAM4, TIGRESS: DREAM4 and 5, Mix-CLR: DREAM3, iRafNet: DREAM4 and 5). This can unfortunately lead to an unification of the methods as they are optimized for the same data-sets and public. Out of the 14 methods only 4 present an application to human regulatory networks: ARCANE (B cells), Regulatory Circuits (394 different tissues), SINCERITIES (monocytic THP-1 human myeloid leukemia cell differentiation into macrophages) and PoLoBag (bone marrow CD34+ cells). Both ARCANE and Regulatory Circuits produced regulatory networks on B cells, the cell type we are looking into.

Addition of regulatory regions Out of all the methods, Regulatory Circuits is the only method to use the regions accessibility information as entry, translated to their activity. As we saw in the previous section, the accessibility of the chromatin has a major role on the regulation as it constrains transcription factor binding to DNA (a regulatory region in a "closed" conformation will inhibit any potential regulation for a TF with a binding site inside it). This information about regulatory regions is also important when looking into perturbations of the regulatory network in cancer, as mutations can occur in non-coding areas and still impact the regulation. Regulatory regions can even be modified in their accessibility or location by pathological processes such as cancer. One way of obtaining this information while not identifying the regions themselves, is to look at the TFs binding sites as COALESCE, DISTILLER and iRafNet do. But they look at TFs binding sites at a given moment and not comparatively on several times points.

Signed networks ans scored ones Few methods (PoLoBag and SINCERITIES) produce signed networks, i.e. specify if the regulation is positive (induction) or negative (inhibition) but most of them produce network with weighted edges (exception of REAVEAL, BANJO and NIR) Some of these methods scores are based on statistical weight: for example in Regulatory Circuits the weight of the binding site confidence is based on the conservation of the TF binding site, meaning a large amount of data is necessary to perform this analysis. COALESCE use a probabilistic approach normalized by the likelihood of the same observation done by chance. The methods inferring signed networks are also the most recent ones, this may be the result of a shift and progression in regulatory network inference methods.

Issues of reusability and reproducibility The closest method to what we are aiming to realise - inferring regulatory network to look at regulatory perturbation in specific diseases - is Regulatory Circuits. It takes as entry regions activities, from which genes activities are approximated, and information on TFs binding sites, the type of information we have. Unfortunately they did not infer regulatory networks on the Follicular Lymphoma and only in one population of normal B cells (CD19+ blood B cells). We can't reuse this network since we are interested in different subsets of B cells which are all either distinct or included in this latter population. Also, as we will develop in later chapters, Regulatory Circuits is not reproducible nor easily reusable.

This is an issue with most of network inference methods: they present algorithms without implementation limiting the reuse of the methods or implementation without standardization of the output files. Data are usually released as primary raw datasets, usable processed data or compiled networks but with few possibilities for easily adding new links between the data or for re-using the published bioinformatics pipelines.

Many of these methods produce local files, often specifically formatted for the current analysis and are rarely designed to be easily available and reusable. To help the reproducibility and the reusability of the workflows of analysis and their results, there is a need to unify life-science databases and to populate them.

Conclusion on regulatory network inference methods We can see that there is a variety of regulatory network inference methods but very few capitalize on the heterogeneity of the data produced in life science, often focusing only on the genes activities. And they often concentrate on bacterial regulatory networks, whom are smaller in size but easier to get more time point for the activities. Few of these methods also provide signed networks, but knowing if a TF is an activator or an inhibitor could be very useful for example to determine what happen to a cell.

The main issue with these methods is the lack of sustainable solution to store and share the produced regulatory networks, which limits their reuse. We could look at existing data-bases to find a solution.

2.3 Data Structure in life science

Current life science data structures There are currently more than 1600 life science databases, each able to answer important questions in a particular domain [RIGDEN et FERNÁNDEZ, 2019]. There has been a long-standing effort to standardize and integrate reference datasets and databases [ALDHOUS, 1993 ; STEIN, 2003]. But, despite these efforts, many studies' data are provided using specific and non-standard formats [CANNATA et al., 2005]. And most of them offer a dedicated repository for expert knowledge but they fail at structuring biological datasets [STEIN, 2003]. Indeed, the classical data management technologies used by the life science community range from data storage in the form of multiple tabulated files analyzed with spreadsheets, silo models in complex database management systems with a predetermined scheme of federated data such as Intermine [KALDERIMIS et al., 2014] or Biomart [SMEDLEY et al., 2015], to *ad-hoc* community centralized models such as in bio-imaging communities. These solutions address immediate integration requirements but they are poorly compatible with scalable and flexible integration needs, either between communities (for example to jointly analyze medical imaging and genomics data) or with the world of linked data to enrich analyses with symbolic knowledge selected in a precise and contextual way in existing databases. This limits the capacity to reuse the studies' data in other pipelines, the capacity to reuse the pipeline's results in other studies, and the capacity to enrich the data with additional information [AL KAWAM et al., 2018].

Some notable and massive databases have been released following effort of standardization and good practices, it is the case with ENCODE [CONSORTIUM et al., 2012] [GERSTEIN et al., 2012], FANTOM5 [LIZIO et al., 2015] [ANDERSSON et al., 2014] and RoadMap Epigenomics [SKIPPER et al., 2015] [KUNDAJE et al., 2015] consortia.

ENCODE (ENCyclopedia Of DNA Elements) is a National Human Genome Research Institute (NHGRI) project and aims to identify all functional elements in the human genome. They were 440 scientists from in 32 laboratories collaborating on the project in 2007. It was developed with the funding a $80 million by the NHGRI in 2007. A recent overview of the content of ENCODE can be found in [SNYDER et al., 2020].

The FANTOM5 (Functional ANnoTation Of Mammalian Genome) consortium aims to generate both a map of the majority of human promoters and transcriptional regulatory network models of each cellular state. It was initiated by the RIKEN institute. It contains RNA-seq, short RNA-seq and CAGEscan data for approximately 400 cell

types.

The Roadmap Epigenomics Mapping Consortium aims to produce a public resource on human epigenomic data. It was launched by the National Institutes of Health (NIH) and is a collaboration of 10 groups, mainly north-American. The current release contains a total of 2,804 genome-wide data-sets (including: histone modification, DNase, DNA methylation, and RNA-Seq).

Another project, proposing a large data base of regulatory networks is Regulatory Circuits [MARBACH et al., 2016]. Regulatory Circuits is based on previous works from FANTOM5 and present a collection of 394 scored tissues-specific regulatory networks. Their goal is to identify disease-associated mutations that impact the regulatory networks.

Unfortunately, those datasets have no or low compliance to the FAIR guidelines [WILKINSON et al., 2016]. ENCODE data for example have only been published as ontologies [MALLADI et al., 2015], processed data together with scripts used to obtain them, or unlinked datasets.

Semantic web technologies An alternative approach for structuring and analyzing heterogeneous datasets and knowledge bases, such as those encountered in Life Sciences, is based on the Semantic Web technologies. They are an extension of the current Web that provides an infrastructure for integrating data and metadata in order to support unified querying and reasoning as a virtual unified data-set [BERNERS-LEE et HENDLER, 2001]. This approach has been widely adopted by the life science community for releasing reference data and knowledge bases [JUPP et al., 2014 ; WHETZEL et al., 2011] in RDF triplestores. Semantic Web technologies have been perceived as a relevant framework for supporting integration [BLAKE et BULT, 2006], and have been widely adopted [ANTEZANA et al., 2009 ; H. CHEN et al., 2012], but some challenges remain to achieve Web-scale integration [KAMDAR et al., 2019]. Thanks to the growth of linked data, supported by the Linked Open Data initiative (LOD) [BIZER et al., 2009], more and more reference data and knowledge bases are integrated. We can see in FIGURE 2.3, the important part of Life Science databases in the Linked Open Data cloud diagram and their dense interconnections. Moreover, it also evolved into the FAIR principles for ensuring that the available data are Findable, Accessible, Interoperable and Reusable [BRANDIZI et al., 2018 ; LIVINGSTON et al., 2013 ; RODRÍGUEZ-IGLESIAS et

al., 2016 ; WILKINSON et al., 2016].

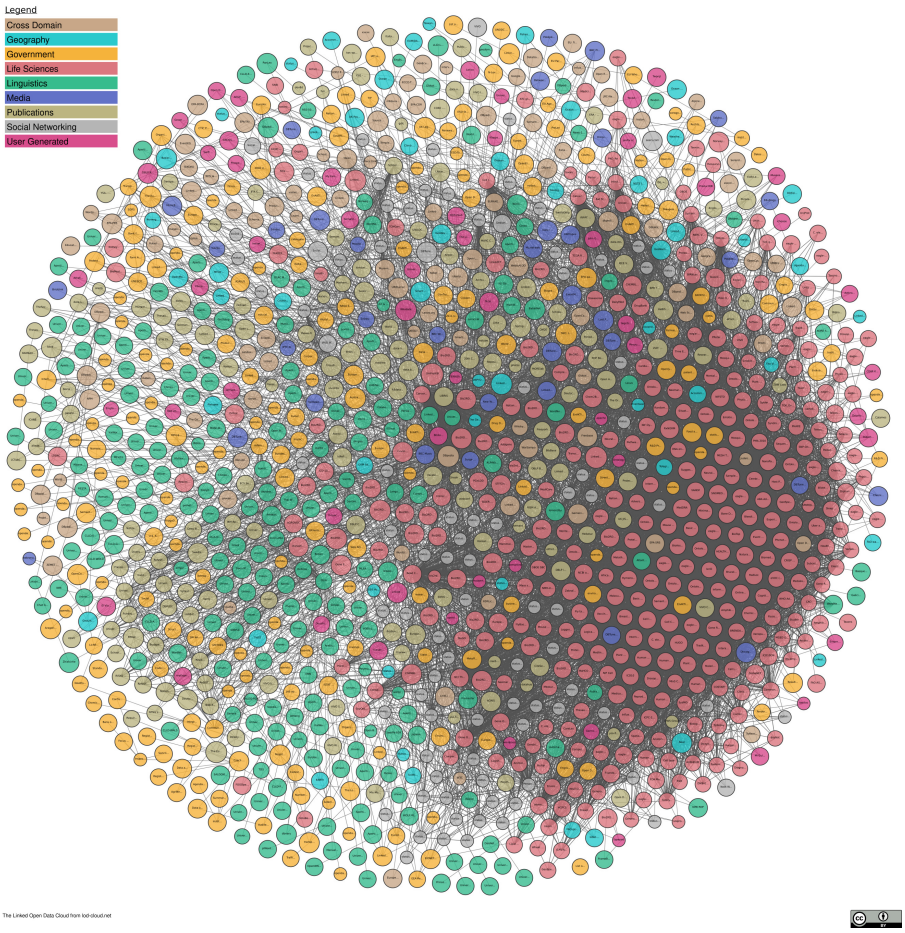


FIGURE 2.3 – Link Open Data cloud diagram as download the 2020-09-10: Life science is represented in dark red.

2

Some notable databases have been released under the RDF format, for example: FANTOM5, although it only concerns gene expressions and not regulatory data [ABUGESSAISA et al., 2016 ; LIZIO et al., 2015].

RDF and SPARQL Resource Description Framework (RDF) is a data model using triples between a subject, a predicate and an object. The data are described as graph using the relation between the different entities. Once described using RDF the data

2. Linking Open Data cloud diagram 2020-09-10, *by Andrejs Abele, John P. McCrae, Paul Buitelaar, Anja Jentzsch and Richard Cyganiak*. http://lod-cloud.net

are integrated using triples-stores such as virtuoso. To query the graph a specific language, SPARQL, is used, an example of SPARQL query from neXtProt [ZAHN-ZABAL et al., 2020] can be seen in FIGURE 2.4.

```
#Proteins which are targets of antipsychotic drugs and expressed in brain
PREFIX drugbank: <http://wifo5-04.informatik.uni-mannheim.de/drugbank/resource/drugbank/>
select distinct ?entry where {
  service <http://wifo5-03.informatik.uni-mannheim.de/drugbank/sparql> {
    select distinct ?unipage WHERE {
    ?drug drugbank:drugCategory ?drugCat.
    ?drug drugbank:target ?target.
    ?target drugbank:swissprotPage ?unipage.
    filter(?drugCat = <http://wifo5-04.informatik.uni-mannheim.de/drugbank/resource/drugcategory/antipsychoticAgents>
    || ?drugCat = <http://wifo5-04.informatik.uni-mannheim.de/drugbank/resource/drugcategory/antipsychotics>)
    }
  }
  ?entry :swissprotPage ?unipage.
  ?entry :isoform /:detectedExpression / :term /:childOf cv:TS-0095. # detected in brain
}
```

FIGURE 2.4 – Example of SPARQL query from neXtProt [ZAHN-ZABAL et al., 2020], extracting proteins that are target for a specific drug in a specific tissue.
[3]

AskOmics A bottleneck for a broader adoption of Semantic Web technologies by the life science community is a technical barrier: a complete analysis scheme based on Semantic Web technologies requires users first to prepare their data according to a RDF framework to make them exploitable, and second to become familiar with the SPARQL language [PÉREZ et al., 2009] either for querying their own RDF data, or for analyzing them in relation with the other triplestores of the LOD.

AskOmics [4] [5] is a visual interface for intuitive data integration and querying, developed by the Dyliss team at INRIA to answer this issue. AskOmics generates two graphs from tabulated files, some format specific data-sets (e.g. BED or GFF format), as well as original RDF. Firstly, the graph of data is a RDF representation of the content of files provided by the user. It is hidden from the user. Secondly and presented to the user, the graph of entities and values types which is a representation of the structure of the graph of data. It is also much smaller. It is used as a visual proxy for allowing end-users to compose SPARQL queries intuitively over the graph of data with a graphical interface.

3. https://www.nextprot.org/proteins/search?mode=advanced
4. https://github.com/askomics/flaskomics
5. https://flaskomics.readthedocs.io/en/latest/

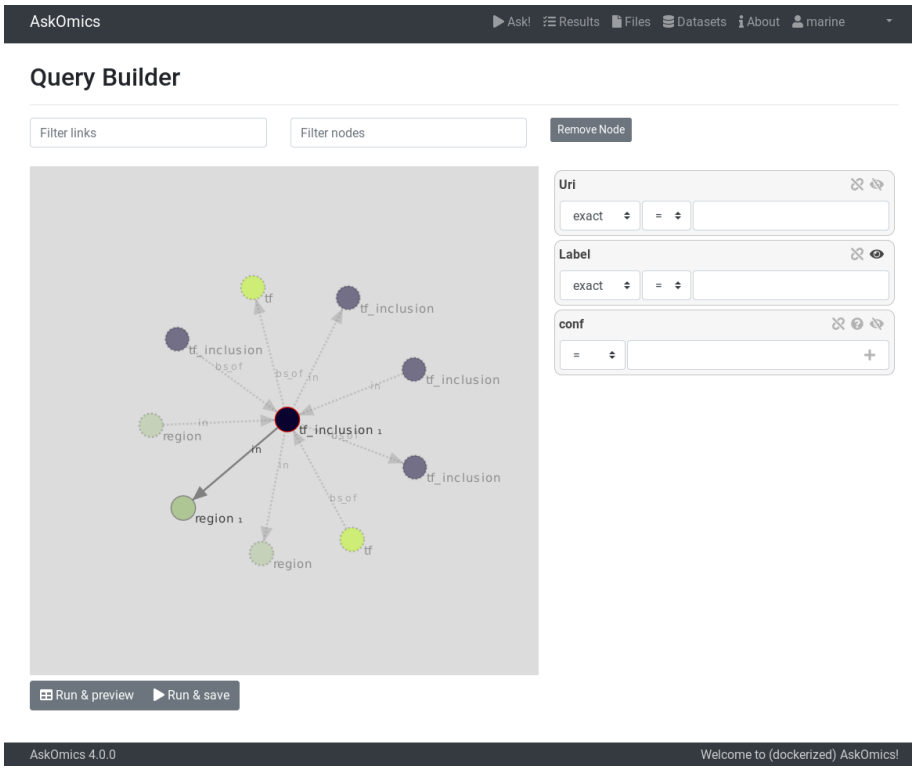


FIGURE 2.5 – AskOmics query builder.

In our works, we use AskOmics as a tool for the integration of the biological data into relevant end-points, it is also used to help create the SPARQL queries to interrogate the RDF graphs.

From workflow to data-graph Semantic Web technologies, is useful to transform workflows as graph-data queries. As an example, Regulatory Circuits, where the workflow follows relations from one entity to another would be a great case study to test this kind of semantic web implementation. The relations between two entities can be interpreted as a Triple and SPARQL allows to compute scores for the relations.

One difficulty of using Semantic Web technologies is the need for clean and formatted files as input. This may require quite a lot of pre-treatment of the data do ensure that the header are well formatted and that the entities have the same identifier in all files and that one identifier is not used for two different entities. This would alos require a selection of relevant information among the provided data-sets, that may not be trivial, as we will describe in latter chapters.

An issue of using Semantic Web technologies is the iterative queries: for example in data-sets such as Regulatory Circuits the networks are inferred in a first time at the sample level, and then those are used to compute the networks the tissue level (union of several samples). This means that we need to query the data a first time at one level, re-inject the result of the query into the triple store and then query it again. Unfortunately, this does not scale well and can be time consuming and lead to several layers of potential mis-computation.

This thesis will focus on finding solutions to identify the parts in regulatory network inference workflow that can be formalized as RDF data graph and queries to obtain the regulatory networks. This also means identifying which part need to stay as workflow: such as the cleaning of the input files, post treatment... We also develop on the named graph as a solution to represent the different level of information computed for obtaining the regulatory networks.

2.4 Synthesis

This section introduces the following chapters representing the different contributions of this Ph.D thesis.

Regulatory circuits and study of reproducibility In CHAPTER 3, we looked at the Regulatory Circuits project, which is amongst the most recent and the most complete attempt to identify human cell-type specific regulatory networks. The underlying goal of this resource is to find disease specific perturbations of the cellular regulatory networks, which is what we are aiming to do on a specific cancer although it was generated on a greater scale with a large public data-set. We looked into applying their methodology to new data-sets to infer novel regulatory networks. Regulatory Circuits method is based on two levels of regulatory networks: one based on samples - experiments - and a second on the tissue-level - composed of at least one sample but often the union of several samples.

We could not directly reuse the provided computed networks as Regulatory Circuits networks aggregate the cell populations we are investigating for the normal differentiation into only one (CD19+ B cells) and do not include any cell type related to Follicular Lymphoma. We also discovered that the implementation provided was not reusable, so we proposed two ways of re-computing *Regulatory Circuits* networks based on the available information on their methodology: one recomputing all the steps when it was possible, applying the methodology described in the paper accompanying the resource and a second one using all the pre-processed intermediary files as entry. But even when following every steps of the described workflow we could not reproduce the published results. These three limitations illustrate three common pitfalls of reproducible science.

This led to the design of a new method for inferring genes regulatory networks, that responded to those pitfalls, but also added signs to the predicted regulatory relations. This new method is described in a later section, corresponding to CHAPTER 6.

Semantic Web technologies as framework to format network inference workflow In CHAPTER 4, we introduce an approach based on Semantic Web technologies to revisit the analysis workflow performed in the *Regulatory Circuits* study to make them more

easily available and usable. We show that the first level of *Regulatory Circuits* analysis workflow can be formalized as two SPARQL queries and that the performances were acceptable: less than four hours to compute all sample-specific networks.

Our approach consisted in structuring the data and results of a systems biology study as a RDF dataset. We produced a RDF model of *Regulatory Circuits* that provides a unified access to their sample-specific networks. Our results showed that once the relations and necessary transformation had been pre-computed, the *Regulatory Circuits* analysis pipeline could be formalized as two SPARQL queries, for the sample-specific networks. But it still requires post-computation to be able to scale to the tissue-specific level.

We argue that using Semantic Web technologies to format networks inference workflow as data-oriented graph is a solution that improves the re-usability of the method and their interoperability with external data. At this step, we only provide the data-graph of the biological data and the query to compute the sample-specific networks, there is no conservation of the computed networks. Also, this approach proved a lot more challenging when scaling to the tissue-level networks.

RDF graphs multi-layered structure for regulatory networks In CHAPTER 5, we extended the previous works on *Regulatory Circuits*, leading to scaling of the RDF data-graph to not only recover sample-specific networks but also tissue-specific ones.

We elaborate upon the strategy to generate a public RDF resource which contains not only the *Regulatory Circuits* source biological data, linked to standard LOD resources, but also the results of the analysis pipeline at the sample and tissue-specific layers. We use the same RDF model of the biological data as described in the previous paragraph and re-compute the same first two steps, but after computing the sample-specific networks we re-injected them in the RDF endpoint in named-graph in order to query them to compute the final step of the original workflow: tissues-specific networks. The resulting networks are themselves also re-injected as named graph in the triple-store.

This part proved more challenging, resulting in an almost one month long computation on the triple-store between computing the queries to get the scored relations and re-injecting their results in specific named graphs representing each network.

A new network inference method In CHAPTER 6, we present a new design of pipeline to infer new regulatory networks, with a focus on small sets of biologically-closely related cells types. This new method also produces signed networks. As mentioned in a previous section, the need to develop a new approach for our specific data arose with the inability the find fine-grain analysis at the level of cell population and not cells tissues that include them all in one network, or the nonexistence of Follicular Lymphoma networks in the current available data-sets. It was reinforced by our inability to reproduce *Regulatory Circuits* method with convincing results.

This new strategy takes advantage of the heterogeneity of the available data it uses: information about the gene expression, information about the chromatin accessibility, TFs activities and their binding site localization. We use a multi-level normalisation of the activities called Pattern, clustering genes and regions of similar activities trajectories across the population. As we focus on smaller data-sets, the analysis is not based on statistical methods but is descriptive. We use a similar structuring in RDF data-graph once the data are normalized as the one described for re-implementing *Regulatory Circuits* and query it using similar queries to get the relations between TF and genes.

The post-processing step is a reasoning method based on expert knowledge to describe the potential relations of regulation in accordance to the biological context and rules. This was designed to be voluntarily stringent as the goal of our method is not to describe all potential interactions but to extract the regulators with the higher confidence of impact on the network. Our method also produce signed networks based on the consistency of the activities of the regions, genes and TFs across all populations given as entry.

We tested the pipeline on *Regulatory Circuits* data-sets and managed to find regulatory relations obtainable according to their data, but with a better recovery rate of lowly expressed genes. We also were able to check the consistency of the sign of the relations with two major databases (Signor and Trrust) and are in accordance for 70% of relations found in both our networks and the databases.

Application to B cells Finally, CHAPTER 7, we applied the newly designed pipeline of the previous section, onto a specific set of four closely-related cell types found in the B cell differentiation: : naive B cells, IgG and IgM memory B cells and plasmablasts, in

order to gain a better understanding of the normal differentiation.

We extracted 314,965 TF-genes relations, resulting in two level of regulatory networks: an overall graph presenting the relation between the different Patterns of genes clusterisation and several specific graph of TF-genes interactions. But the graph are highly dense, so we proposed a way of reducing the number of regulation master candidates. The goal is to be able to produce a list of selected TFs in order to biologically test their impact on the networks to validate their importance. We chose two criteria: coverage - the ability of the TF to have a large amount of targets - and the specificity - the fact that the TF mainly regulate only a certain direction.

Using those criteria we were able to produce a list of 146 TF that pass them and seems to be key regulatory in at least one pattern of this differentiation. This list includes BACH2, PRDM1, PAX5 and IRF4 four of the five regulators on which we had literature knowledge about. The only missing TF from the literature is BLC6, which was unfortunately not present in our data-set as a potential TF and thus could not be retrieved.

These TFs will need to be confirmed through biological experiments but our pipeline allows to reduce the search space.

CHAPITRE 3

REGULATORY CIRCUITS AND ITS LIMITS

The regulatory circuits project is amongst the most recent and the most complete attempt to identify cell-type specific regulatory networks. Its goal is close to what we were aiming to do, on a higher scale and with very large public datasets. This chapter goal is to determine in which ways can *Regulatory Circuits* be used to infer new regulatory networks.

SECTION 3.1 completes the biological introduction presented in CHAPTER 2 and presents the biological context supporting *Regulatory Circuits*. SECTION 3.2 details the workflow and the result files of *Regulatory Circuits* as 394 tissues-specific networks described by scored TF-genes relations. SECTION 3.3 analyses the limits of the methodology and networks of *Regulatory Circuits*. In the original *Regulatory Circuits* article, several steps of the workflow were under-specified. We conducted a reverse-engineering of their intermediary results. As the implementation provided was not running, we propose two new implementations of their method, presented in SECTION 3.4. These 2 strategies for computing the relations score are compared to the original *Regulatory Circuits* results. We could only obtain similar (but not identical) results to *Regulatory Circuits* on subsets of the relations.

We discovered that (1) the workflow is not reusable, (2) even when re-implementing the workflow we can not reproduce their results and (3) *Regulatory Circuits* results are too coarse, as they aggregate the 4 cells population we want to compare into a single network. These three limitations illustrate three common pitfalls of reproducible science.

3.1 Introduction

The *Regulatory Circuits* project [1] [MARBACH et al., 2016] is one of the largest effort of genomics data integration in human cells and consists of several analyses on heterogeneous and multi-layer "omics" data on 394 human cell lines and primary cells from tissues.

The *Regulatory Circuits* website gives access to unstructured, disconnected and diversely formatted tabulated files related either to source biological data (FANTOM5 data, genes and regions genomic coordinates, TFs binding sites occurrences... divided in 26 files), to computation intermediate results (59 files), or to the results of in-silico integrative analyses (394 files, one for each network).

The outputs of the project are only available as text files on their website, this has huge impact on i) the reproducibility of the results, ii) their maintenance as they will need to be updated when newer or additional data sources are released and iii) their reuse for advancing other studies (which was the reason these results were generated in the first place).

This project is a major provider of biological data, cited more than 157 times (Google Scholar). its resulting networks were used in at least 42 other articles.

3.2 What is *Regulatory Circuits*: biological model, input / output data and computational concept

FIGURE 3.1 presents the biology behind regulatory circuits. The interaction between a TF and a gene is determined as the ability of the TF to bind in a region close the gene. From a biological perspective, the regulation of a gene by a TF results from two mechanisms, as TFs can bind to two types of regulatory regions called promoters and enhancers. In *Regulatory Circuits*, the authors place on the DNA the isoforms of the gene (assimilated to transcript for the rest of this chapter), the relation is calculated between the TF and the transcript then the transcript is linked to its corresponding gene.

The entry data used in *Regulatory Circuits* comes from several independent pro-

1. http://regulatorycircuits.org/

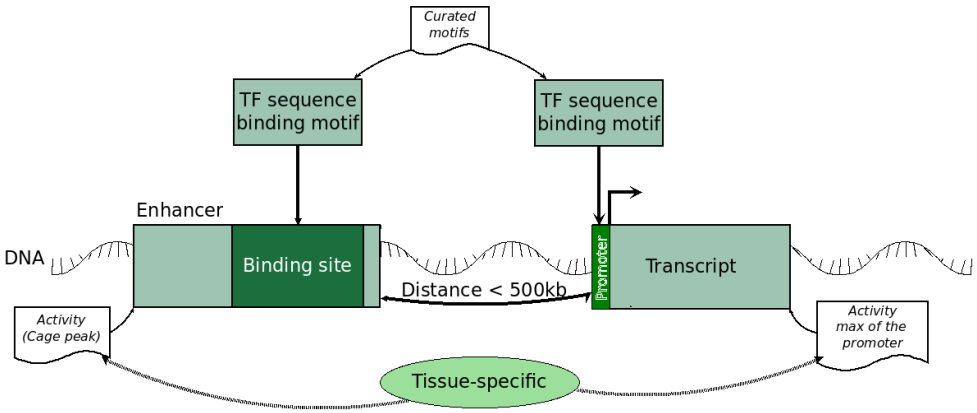


FIGURE 3.1 – Biological principles behind *Regulatory Circuits*

jects: FANTOM5 [ANDERSSON et al., 2014] for measuring transcription or regulatory activity (using CAGE-peak data on the regulatory regions, 808 different samples having information for both enhancers and promoters), ENCODE [CONSORTIUM et al., 2012] for the prediction of transcription factor binding sites, and GTex [LONSDALE et al., 2013] and Roadmap Epigenomics [SKIPPER et al., 2015] for validation data.

Regulatory Circuits computes networks that are not derived from a statistical analysis of biological measurements but based on a set of computed correlations between regulatory regions activities, gene expressions, and curated and scored TF binding sites.

Datasets were published either as input (raw data) or intermediary (authors- processed) data files, in the form of tabulation or comma-delimited data files with various formats and contents.

The output of Regulatory Circuits study is a set of 394 scored tissue-specific regulatory interaction networks that can be explored through text files, representing 9.1 GB. Each network can be described by an oriented graph in which TFs are connected to genes by weighted edges.

3.3 What is *Regulatory Circuits*: Detailed workflow

FIGURE 3.2 presents an overview of the *Regulatory Circuits* workflow, each steps is described in the following subsections.

FIGURE 3.2 – *Regulatory Circuits* global workflow and its different steps.

3.3.1 Global formula

The score ($w_{ij}(S)$) of a relation between the transcript (j) and the TF (i) in the sample S is based on the distance weight(d_{jk}) between the transcript and the regulatory region (k), the confidence score of the binding site of the TF in the given region (c_{ik}), the normalized activity of the region (x_k) and the normalized activity of the transcript (y_j). Giving the following formula:

$$w_{ij}(S) = c_{ik} \times d_{jk} \times \sqrt{x_k(S) \times y_j(S)} \tag{3.1}$$

For Promoters, the distance is normalised to 1 -as the promoter is adjacent to the transcript- and the transcript activity is approximated by its promoter activity. The previous formula is then reduced to :

$$w_{ij}(S) = c_{ik} \times x_k(S) \tag{3.2}$$

3.3.2 Normalized expression activity of regions and transcripts

As described in the previous section, the activities of the different elements are normalized.

As shown by the equations 3.1 and 3.2, the choice of the normalization is important in such workflows as the value directly influences the score of the relation between the TF and Genes in the resulting networks.

How ranks were described in the method section

In *Regulatory Circuits* the authors describe the transformation of the activities of the different regions as a normalization from 0 to 1, and that the normalization is done regulatory element by regulatory element:

> *The weight of promoter-gene edges was defined as the normalized activity level of the promoter across all samples (normalization was done per regulatory element because expression levels of diverse enhancers and promoters might not be on the same scale). Thus, if the promoter is not active in a given cell type, the edge weight is 0 (i.e., the edge is not present), and if the*

> *promoter is maximally active, the edge weight equals 1.* [MARBACH et al., 2016]

For the isoforms (transcripts), the activity was based upon their promoters:

> *The activity level of isoforms was defined as the maximum activity level of their promoters (which are usually few—the majority of isoforms have only one or two alternative promoters).* [MARBACH et al., 2016]

In the article, the normalization function used is not further described.

How ranks were really calculated

Using reverse engineering, we managed to find out for the enhancer that the normalization is an application of the rank function, this information was given in the name of the transformed expression files: *enhancer_exp.rank.txt*, *promoter_expr.rank.prec90.txt* and *transcript_expr. rank.prec90.txt*.

For one enhancer the expressions of the different samples are ordered from the least expressed to the most expressed. Several samples can have no expression and their ranks are set to 0. The other samples ranks are calculated as: Position in which they appear after being ordered divided by the number of samples expressed. For an enhancer, the most expressed samples are therefore normalized to 1.

For the promoter, when reverse-engineering the rank, we realised that the ranks were not calculated element by element in this step. Instead all promoters were regrouped based on the transcripts they were preceding, and the rank function was applied to all the expression for the samples of the promoters, i.e. if a transcript is preceded by 2 promoters the ranks of the promoters were calculated on 2 time 808 samples.

For the transcripts, the rank for a sample was supposed to be the maximum rank of its different promoters, but it was not what we found in the intermediary files. The transcripts' ranks were computed as the means of its promoters ranks weighted by a different constant for each transcript. Therefore we could not undoubtedly compute the transcript scores.

3.3.3 Confidence score of the TF binding sites

In *Regulatory Circuits*, the authors consider 662 TF and their matrix of binding sites in the genome. They used a curated collection of matrices and assigned a confidence score to each binding site based on the conservation across mammals using the works: [KHERADPOUR et al., 2007][KHERADPOUR et al., 2013][KHERADPOUR et KELLIS, 2014].

The binding sites were looked for in a 400 bp upstream to 50 bp downstream window of the promoter considered, and limited to the actual chromosomal coordinates of the enhancers.

If several binding sites of a same TF were found in a regulatory region, only the maximum of their confidence scores was kept for the TF-region relation. The scores and relations between TF and regions are compiled in the *tf- - -promoter. prec90.txt* and *tf- - -enhancer. prec90.txt* files.

3.3.4 Distance weight of the regions

The weighting function for the distance only applies to the enhancer, for the promoter the weight of the distance is set to 1.

To normalize the distance between enhancer and transcript, the authors used cis-eQTLs from RegulomeDB [BOYLE et al., 2012] and computed their distance to the TSSs of target genes. The weigh function is defined using a local polynomial regression fitting for the range 1 kb to 500 kb (in either direction from the transcript), where 1 kb was normalised to 1 and 500 kb to 0. This is then applied to the enhancers considered in the workflow.

This means that all enhancers further than 500 kb of the transcript were not considered for the remainder of the computation.

Using the computed files *enhancer- - -transcript. prec90.txt* and *promoter- - -transcript. prec90.txt* where the computed weighted distance are given we were able to recalculate the polynomial regression and to apply it.

3.3.5 From individual relations to networks

Up until this point all relations are calculated using the transcripts (i.e. isoforms) instead of genes. To find the TF-genes relations, all relations between a TF and isoforms of a same gene are combined into one and score as the maximum of all those relations.

Similarly, if a TF-gene relation exist using several promoters (i.e. enhancers) the relation is kept using the maximal score computed.

> *For each pair of edges forming a chain that connects a TF to a promoter to an isoform [...]. If several redundant edges between the same TF and gene were found (via different promoters or isoforms), they were merged and the maximum edge weight was retained. A separate TF-gene network encapsulating all regulatory interactions via enhancers was created using the same approach.* [MARBACH et al., 2016]

At this point, the *Regulatory Circuits* workflow gives two ways of calculating a TF-gene relation and its associated score: one using the enhancer and the second using promoters. In the resulting networks this distinction is not made and the two ways of computing have been combined:

> *Both TF-gene networks thus had edge weights ranging from 0 (absent edge) to 1 (highest confidence), which were added to form a combined TF-gene network including evidence from both promoters and enhancers.* [MARBACH et al., 2016]

As the intermediary files for the networks computed by enhancer and promoters were not available, we were not able to confirm the "added" notion and in which way the score are combined to produce the final networks.

3.4 Issues with *Regulatory Circuits*

While trying to run and understand the *Regulatory Circuits* workflow we run into several setbacks as summarized in FIGURE 3.3. As mentioned in the previous subsection, some of steps of the workflow were under-described or not corresponding to what was in the intermediary data.

FIGURE 3.3 – *Regulatory Circuits* global workflow as presented in FIGURE 3.2 with the addition of the point of contention: either issues in the workflow or remaining doubt on the step.

3.4.1 Understanding the files

A first setback arose during the inspection of the files available for download. It was related to the difficulty to explore them and to extract the information they contain. Some of the files have non explicit names regarding what they contain (e.g.: *hg19. cage_peak_tpm. osc. txt* which contains the expression of the promoters). Not all the files are in the same format: some were comma separated, others, tabulation separated. 15 of the files had headers, while 6 did not, moreover, two of the files presenting header had misaligned header to data. And some files start with a list of comments before the data, up to 1700 lines.

It was also difficult to link the data between two files as the identifiers referring to the same entity were not consistent among the files (e.g.: chr:start-end in file *permissive _enhance. bed* corresponds to *e@chr:start-end in tf- - -enhancer. prec90. txt*).

3.4.2 *Regulatory Circuits* scripts are not usable

The computational scripts and algorithms given as resources are limited to the considered data-set. The given implementation was made using Java but is unusable as such as it lacks explanation on the input files necessary and on how to run it. Furthermore, there is a lack of documentation on the implementation, the wiki is still under construction [2] and have not been updated since 2015. The project website [3] has not been updated and the last news on the project were from august 2016.

3.4.3 Intermediary files not present

While most of the pre-processed data are present in the download folder, the authors do not give any access to some of the intermediary steps of the workflow. The intermediary networks by different type of regulatory regions are not given, nor are the intermediary sample-specific networks. This leaded to issues while trying to reverse-engineer the *Regulatory Circuits* workflow as we could not check the computed intermediary scores before the last step, which leaves several steps unknown.

2. https://github.com/marbach/magnum-app/wiki
3. http://regulatorycircuits.org/

3.4.4 *Regulatory Circuits* methodology could not be reproduced

We ran into several issues with points of the workflow while trying to understand it: the normalisation used for the expression was under-specified, as was the notion of 'addition' of enhancer and promoter network ("Both TF-gene networks[...] were added to form a combined TF-gene network") and some steps were not as described: the rank of the promoter was not calculated element by element. Also, the rank function used as normalization of the expressions only take their order into account not the distance: i.e. 0.1 and 0.6, and 0.6 and 0.7 will have the same distance in Rank (1/nb samples). It is not a fine grain normalization function. Removing or adding samples can highly change the rank distribution: focusing of sub-part of *Regulatory Circuits* means that computed ranks will be different from the ones provided by *Regulatory Circuits*, i.e. the latter are not reproducible. We hypothesised that the rank of two close samples are really close when put with a lot of other samples, and would be wrongly separated when put with very few others.

We also found inconsistencies between the described methodology and the result files. For the ranks of the promoters the formula was not applied element by element as described but on all the promoters of a same transcript at the same time. Also we managed to found regulation between a TF and a gene (INSM1 regulating AMER1) were the gene could only be regulated thought one specific promoter (no enhancer): the relation is scored (4.21371298E-03) in the Myeloma cell line, but the confidence score of the TF in this promoter is given at 0 and applying the formula (3.2) would put the global score to 0 too. In FIGURE 3.4 we show the disparity between the relations found in the result files given by *Regulatory Circuits* and the relations we were able to find by computing the relations between entities across all files.

Moreover, feedback from the Regulatory Circuits' authors was non-existent when solicited about the methodology.

3.4.5 Conceptual issues

Regulatory Circuits stops at the tissue level in the original computed networks, while some users may want to look at finer level such as samples of a same tissue. The provided method therefore lacks some plasticity to have fine-grained / personalized networks.

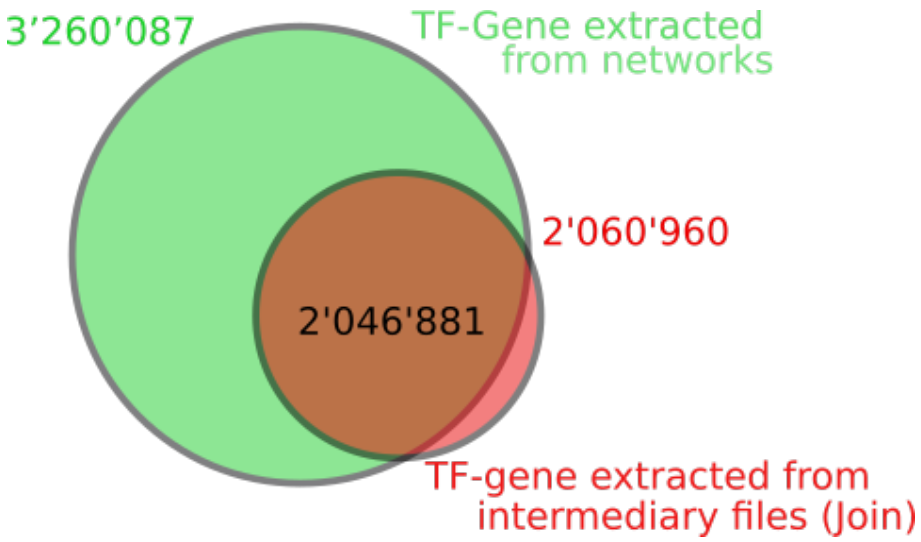


FIGURE 3.4 – Number of unique TF-genes relations found in results network (in green) and number of relations found while following the files (red). The relations in green but not in red can be explained by the lack of some TF in the files. And the relations in red but not in green could be explained by relations always scored to 0.

The way of computing scores seems highly in favor of activation, as they are calculated by multiplying the maximums of several parameters (activities, confidence scores, distance scores...). For the tissue-specific networks, the relations are therefore unsigned and potential inhibition are either hidden or lost. For the same reasons, genes which are less expressed in one sample / tissue are more susceptible to be excluded from a network, disregarding their potential function (for example, TF coding genes are known to be expressed a low to moderate levels).

Another issue with the methodology is that the networks assume the TF to be present in the cell type, the workflow never check if it is really the case. All relations found are assumed to be applied but the TF could be inactivated in the cell type, thus the regulation non-existent.

3.4.6 Problem with re-usability and application to new data/Fair?

The output format of the networks -Text files- makes it impossible to explore and enrich the data by combining them to additional knowledge on entities stored in LOD public databases.

The workflow design makes it difficult both to extend *Regulatory Circuits* by adding

new data or updating them, and to reuse only part of *Regulatory Circuits*, as it require to re-compute all the ranks for all the entities. Depending on the entities the user have, it can also be hard to discriminate the different type of regulatory regions or even to map new regions on the existing ones. It is also difficult to enrich with new type of information, adding genes' expression would mean defining a new formula for scores, as the promoter can not be approximated by the gene (several promoters of one transcript and several transcripts for one gene).

3.5 Computing *Regulatory Circuits*

Regulatory Circuits is a general resource on regulatory networks. We wanted to use it on our specific cells-types, unfortunately the resulting networks of the original study were on a larger scale than the cell-populations we were looking at. The four populations (Naive B cells, Memory B cells (IgG and IgG and Plasmablast) we were interested in were combined in a larger cell-tissue (CD19+ B Cells).This prevented us to discriminate them. A solution would be to run the *Regulatory Circuits* workflow on the new data, but this led to several setbacks and eventually led to our inability to re-compute the original graphs and their scores as described in this section.

3.5.1 Three ways of computing *Regulatory Circuits*

For running *Regulatory Circuits* the first step was to clarify the necessary files and homogenize the identifiers for all entities across the files.

We released two implementations of *Regulatory Circuits*, in addition to their own: the one using semantic web technologies and the given pre-processed files (detailed in CHAPTER 4) and a second one using bash, recomputing the steps we could by following the method described in the article and using the less-processed input files.

In the first implementation we simply reused the rank given in *Regulatory Circuits* as we could easily reuse the intermediary files. We used the computed confidence score of the binding sites and the weight for distance. We also used the explicit links between the entities as given in the various files.

In the second implementation we re-computed the ranks as described in *Regula-*

tory Circuits. We used the R rank function separately on each regulatory element as the principle is the same for both enhancers and promoters. For the transcripts, we took the rank of their promoter and for each sample kept the highest rank across all promoters.

For both implementations, the remainder of the steps of the workflow are computed using the intermediary files of the original study (confidence score, distance, relations between entities, etc.). For the final steps - computing the scores - we applied the formula extracted from the paper, and kept the maximum score for a TF-genre relation: the maximum between the score by promoter and by enhancer and the maximum score in all the samples constituting the tissue.

TABLE 3.1 compares the different steps of the three ways of calculating the resulting tissues-specific networks.

3.5.2 Comparing the three ways to calculate *Regulatory Circuits* circuits

We compare 3 ways of computing *Regulatory Circuits*: the original study using directly *Regulatory Circuits* output network files (1), an intermediate solution using the pre-processed ranks and files of *Regulatory Circuits* but applying the remainder of the pipeline as we understood it (2) and a solution only using input files and computing all the steps (3).

The comparison is done on 12 cells-types: B lymphoblastoid cell line, brain fetal, CD4+ T cells, CD8+ T cells, CD34+ stem cells-adult bone marrow derived, colon adult, colon fetal, epitheloid cancer cell line, pancreas adult, peripheral blood mononuclear cells, small intestine adult and small intestine fetal. We used those 12 tissues as they were part of the 40 tissues on which we had RNA-seq data (from Roadmap epigenomic) to run similar validation of the networks as done in the original paper.

Networks topology

In TABLE 3.2 we show the variability of the regulatory networks depending on the three strategies. In both computed networks, we lost an average of 50 TFs in the result graphs but retained a similar number of genes and lost nearly half of the relations in

TABLE 3.1 – Comparison between the 3 ways of calculating *Regulatory Circuits*. We: scores for sample-specific network using only enhancers, Wp: scores for sample-specific network using only promoters

	OG network (1)	**Using Ranks (2)**	**re-computing Ranks (3)**
Normalised expression Regions	Using the given file	Using the given file	Recomputing the ranks: Done element by element.
Normalised expression Transcript	Using the given file	Using the given file	max(Rank (Promoters))
Confidence score	Given file	Given file	Given file
Distance enhancer-transcript	Given file	Given file	Given file
Link TF-region	Given file	Given file	Given file
Link region-transcript	Given file	Given file	Given file
Link transcript-gene	Given file	Given file	Given file
Score We, Wp	$\emptyset$	Using formula (3.1 and 3.2)	Using formula (3.1 and 3.2)
Score by samples	$\emptyset$	max(We,Wp)	max(We,Wp)
Score by tissues	Result files	max(Samples)	max(Samples)

FIGURE 3.5 – *Regulatory Circuits* global workflow, circled are the files reused and the steps re-computed of the workflow. Circled in red is the first solution which use the given ranks (2) and circled in blue is the second re-implementation in which we recompute the ranks (3).

the final networks.

TABLE 3.2 – Comparison between the 3 ways of calculating *Regulatory Circuits* on 12 cells types. Both method using or re-computing ranks produce networks that are included in the original *Regulatory Circuits* networks.

Type of Circuits	**Given as Result**			**Using the RC rank**			**From scratch**		
	min	max	mean	min	max	mean	min	max	mean
Nb of TF	643	643	643	592	596	594	593	596	594
Nb of Genes	11,911	14,850	13,067	10,902	14,378	12,220	11,881	14,812	12,933
Nb of Relations	407,056	1,796,098	1,042,839	154,354	653,491	414,173	239,049	1,010,008	581,711
% of complete graph	5.2	21.9	12.5	2.3	8.9	5.7	3.3	13.3	7.6

Regulatory Circuits compare the genes regulated in the tissue-specific networks to the expressed genes of the RNA-seq of the Roadmap Epigenomics project. The authors conclude that, as expected, the highly expressed genes are largely (more than 90%) recovered in the produced networks and that the least expressed genes have no regulatory input (less than 10% recovered).

We did the same analysis on the networks found with the three ways of computing (FIGURE 3.6). We found that the original networks and the networks with the re-computed ranks had similar level of recovery of the genes, re-computed network being slightly lower. The re-computed network using the original ranks were even slightly lower than the two others, but it was not significant, except for the bottom 10% of the RNA-seq genes were the percentage found in the network was almost half (9% vs 16 and 15%). But, as concluded by the author of *Regulatory Circuits*, this is a category were we do not expect to recover genes as they have no / low regulatory input.

All relations between TF and gene found in both re-computed methods were relations conserved from the original study networks. The two method developed allow us to retrieve sub-part of the original study networks. As shown in FIGURE 3.7, both re-computed networks are part of the original network and the networks using the original ranks are themselves sub-part of the networks using re-computed ranks.

Scores distributions

As presented in the previous subsection the re-computed networks have a different topology from the original study, and are sub-part of the original networks. But the

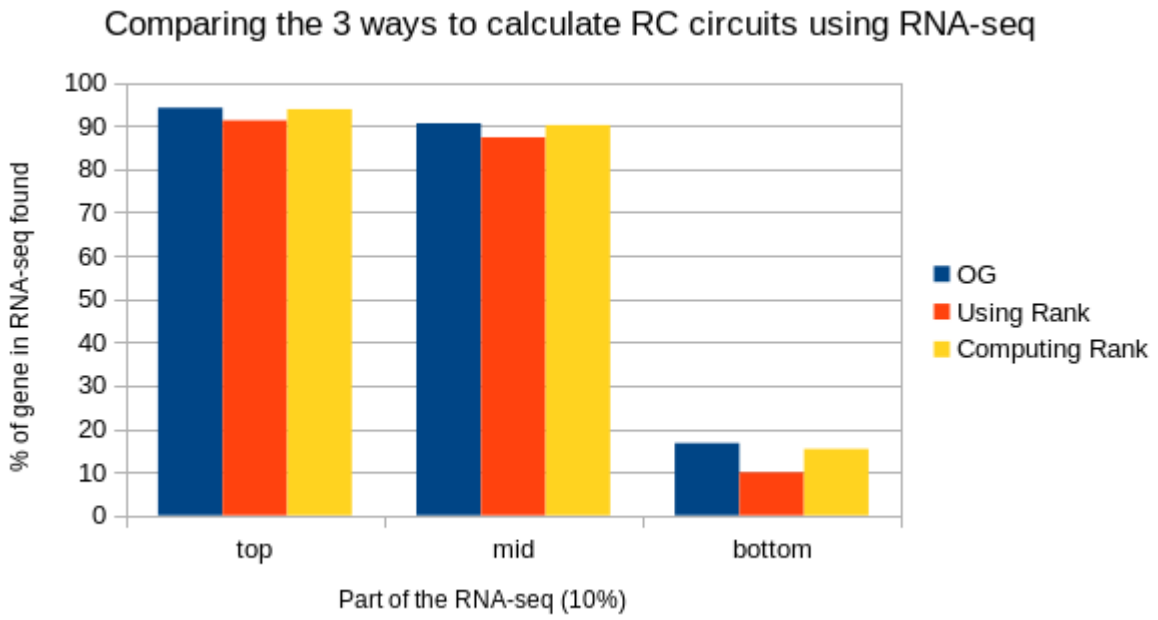


FIGURE 3.6 – Percentage of genes from the RNA-seq related to the networks found in the resulting networks. The RNA-seq genes are separated in three categories: the top 10% most expressed, the middle 10% and the 10% least expressed. Each color represent one way of computing the tissue-specific networks. Analysis done on the 12 networks.

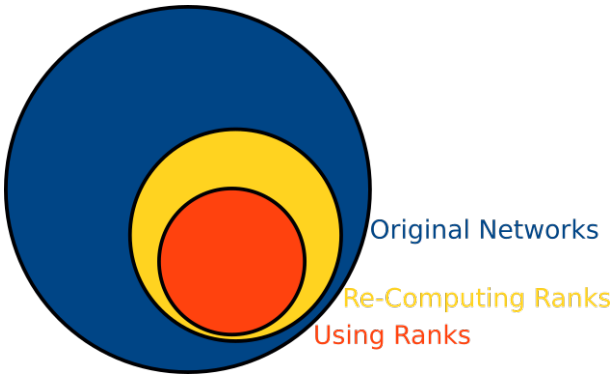


FIGURE 3.7 – Venn diagram of the relations found in the 3 computed networks. Networks recomputing ranks are included in the original networks and networks using original ranks are included in the recomputed ranks ones. The size of the circles are proportional to the number of relations found in the networks.

results of *Regulatory Circuits* were not only the relations but also the associated scores
.

We looked at the score computed in the three versions of the networks (FIGURE 3.8): the score found in the original *Regulatory Circuits* study are in average 10 times lower that the re-computed scores. The computed version with original ranks have slightly higher score than the re-computed ranks one.

This raised the issue of the combination of the enhancer and promoter networks scores, as the original study has scores going over 1 and our method does not. We choose to unify the two networks by a max, but to go over 1 we could have add the scores obtained in the two methods for a relation. But the scores are already higher using the max and the addition would have further augmented the scores. We decided to keep the max in the formula as it was consistent with the other steps of the workflow.

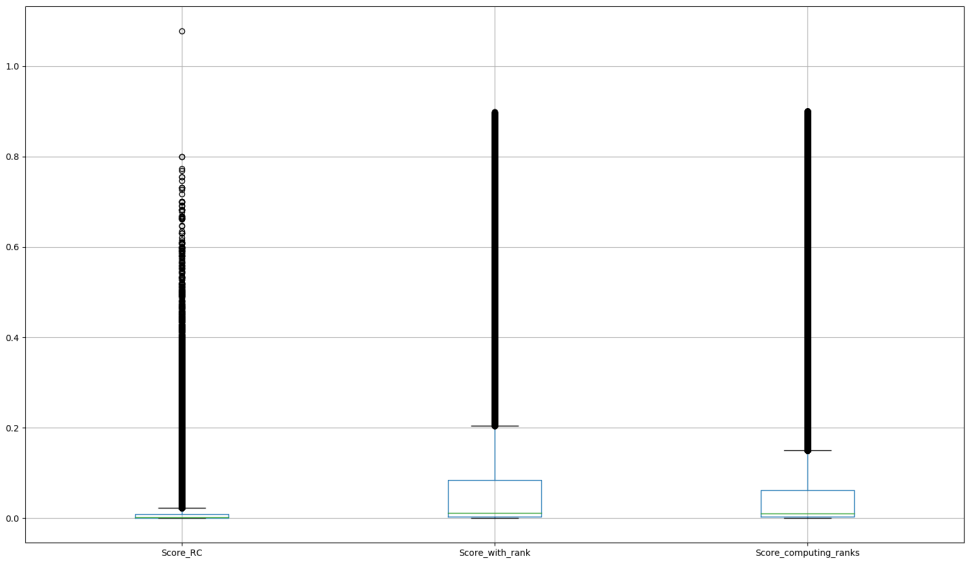


FIGURE 3.8 – Distribution of the scores across the three methods of calculating *Regulatory Circuits* networks. Focus on B lymphoblastoid cell line. Scores in the original network: min 2.82E-7, max 1.08 and mean 0.01. Scores using the ranks: min 2.18E-6, max 0.90 and mean 0.10. Scores when re-computing the ranks: min 4.45E-6, max 0.9 and mean: 0.08

Since the score distribution seemed to vary between the original networks and the recomputed ones, we computed the correlation between the scores. FIGURE 3.9

shows that there is a correlation between the scores in the original study and in the re-computed ones ($r > 0.5$). The correlation in the network using the original ranks is slightly better. This is also the case in the other networks observed: for example in the brain networks r=0,556 re-computing the ranks and r=0,614 using the original ranks. But the correlation between both re-computed networks scores is over 0.85 in all the twelve networks. Making them closer to each other that to the original networks, which hints to other differences between the original workflow to what we applied and was described.

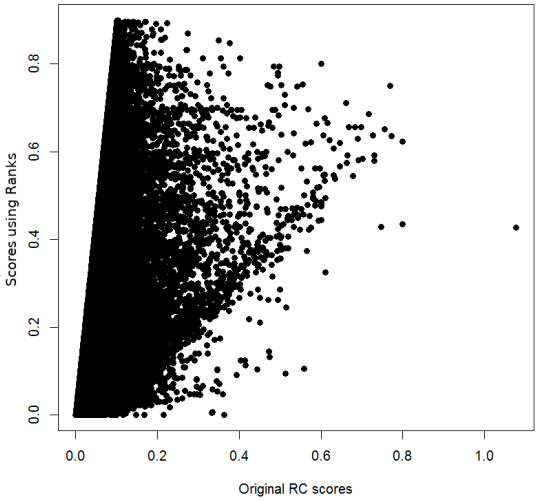
As the re-computed networks are sub-part of the original networks and have overall higher score, we looked into the distribution of the score in the original networks for the conserved relations: for each relation in a re-computed network we went to look at its score in the original network of the same tissue. We found that the relations kept tend to have slightly higher score in average than the relations excluded in both recomputed networks (see FIGURE 3.10). The relations kept while using the original scores are non-statically higher than the one re-computing the ranks.

3.6 Conclusion

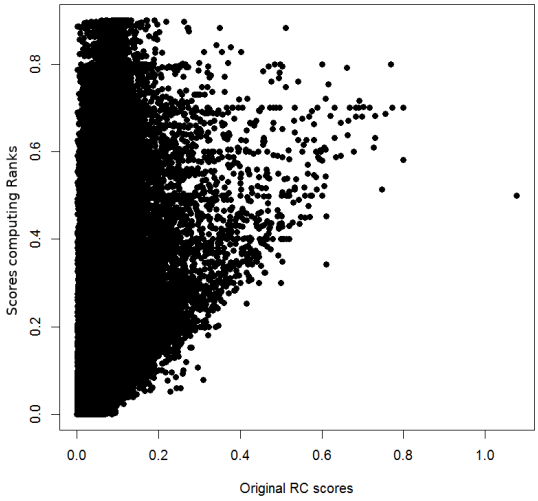
We showed two ways of re-computing *Regulatory Circuits* networks based on the available information on their methodology: one recalculating all the steps when it was possible, applying the methodology described in the paper accompanying the resource and a second one using all the pre-processed intermediary files as entry. The two methods have their advantages: re-computing ranks allows to find larger networks (i.e. with more relations) and retrieves more genes known as expressed from the RNA-seq, but using the original ranks allows to find networks with closer relations' scores from the original study and retrieve relations with an higher confidence (score). We choose to use the second implementation for the remainder of the project, as we wanted to be closer to the original networks.

Regulatory Circuits is a powerful resources of regulatory networks, but it does not provide the fine-grain level we need for our specific cell populations. And it is heavily in favor of activation interactions, and when recomputing it this preference is increased when using the original ranks.

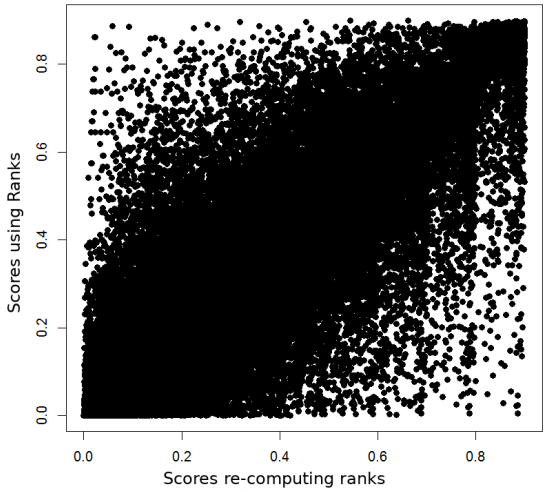
A solution would have been to re-run *Regulatory Circuits* workflow onto our data,



(a) Correlation between original scores and scores of the re-computed networks using original ranks.



(b) Correlation between original scores and scores of the re-computed networks using re-computed ranks.



(c) Correlation between scores of the re-computed networks using re-computed ranks and of the re-computed networks using the original ranks.

FIGURE 3.9 – Correlation between the scores found in the original networks and the scores found in the re-computed networks for the conserved relations. Focus on B lymphoblastoid cell line. The re-computed networks with original ranks have a r of 0.705, the re-computed rank networks have a r of 0.696. Correlation between the two re-computed ranks have a r = 0.928.

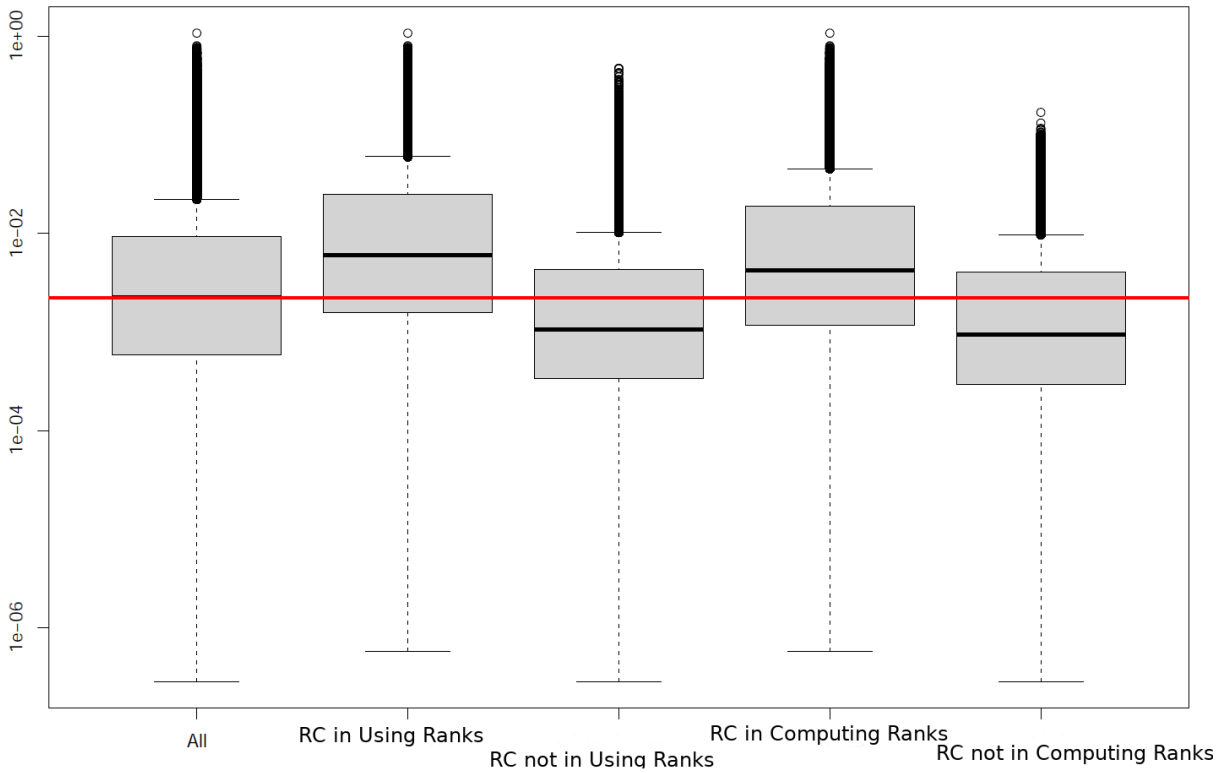


FIGURE 3.10 – Distribution of the scores across the three methods of calculating *Regulatory Circuits* networks, depending of the conservation of the relations from the original network. Focus on B lymphoblastoid cell line. Score presented as log10 of the computed scores.

but as shown previously we were unable to re-compute the original networks even using all the pre-processed data available. This inability to execute the workflow on the original data motivated the need to develop a new pipeline better suited for data-sets composed of close and sparse samples.

CHAPITRE 4

INTERPRETATION OF REGULATORY NETWORK INFERENCE PIPELINE AS GRAPH-BASED QUERIES

This chapter is based on the a article: [LOUARN et al., 2019]

In this chapter, we introduce an approach based on Semantic Web technologies to revisit the analysis workflow performed in the Regulatory Circuits study to make them more easily available and usable.

In SECTION 4.1 we present a structuring of *Regulatory Circuits* using Semantic web technologies and transforming the original workflow as a data oriented graph. We explain in the first step how we identified the relevant files from the original *Regulatory Circuits* data-set. We then map out those files to extract the underlying structure of interaction between the entities they represent. In a third time, we format the existing and relevant files, in order to integrate them. We produce an overview of the integrated files and the underlying data-graph supporting them. In a fourth time, we designed two queries to recover the TF-genes relations through two types of regulatory regions. And finally, we review the performance of this new structuration. Lastly, SECTION 4.3 is a discussion and a conclusion about our approach benefits and limitations.

4.1 Introduction

We propose an RDF representation of the unstructured data files in order to exhibit the chain of relations between the entities involved in the regulation of gene by TFs. To this end, we identified the useful data and we structured them according to a schema supporting the network building task. Based on this RDF representation of the dataset,

we show that the intermediary output of the Regulatory Circuits study can be obtained by two SPARQL queries.

FIGURE 4.1 reintroduce the *Regulatory Circuits* workflow presented in the previous chapter with an en-phase on the steps re-computed during this chapter: leading to the computation of the two missing intermediary files (networks by promoters and networks by enhancers).

4.2 Contribution

Semantic Web technologies provide a generic infrastructure for integrating, combining data with knowledge bases and querying them. They have been successfully applied on reference data, that are arguably the most prone to be reused. We have seen that this requirement also applies to research results, such as the ones from the Regulatory Circuits study. There are some ongoing efforts in the neuroimaging community to use Semantic Web technology for sharing and reusing datasets [MAUMET et al., 2016], but these are not directly applicable to our situation.

By structuring and integrating the data from Regulatory Circuits we aimed at efficiently recovering the TF-gene relationships computed in the original work. We also wanted to make the data structure easily extendable to new data for the users. To do so, a requirement was to identify all the necessary entities from the published datasets (files) and the relevant steps of the pipeline necessary for deriving the relations between transcription factors and genes. We reused the rank values for the expression measures because the Regulatory Circuits' method section does not specify how to compute them, as well as the distance values between enhancers and transcripts. We computed the other elements.

4.2.1 Identifying relevant files among all Regulatory Circuits resources

The first step of data structuring was the identification of all the necessary files from Regulatory Circuits (available from the supplementary archive file [1]), including raw

1. http://regulatorycircuits.org

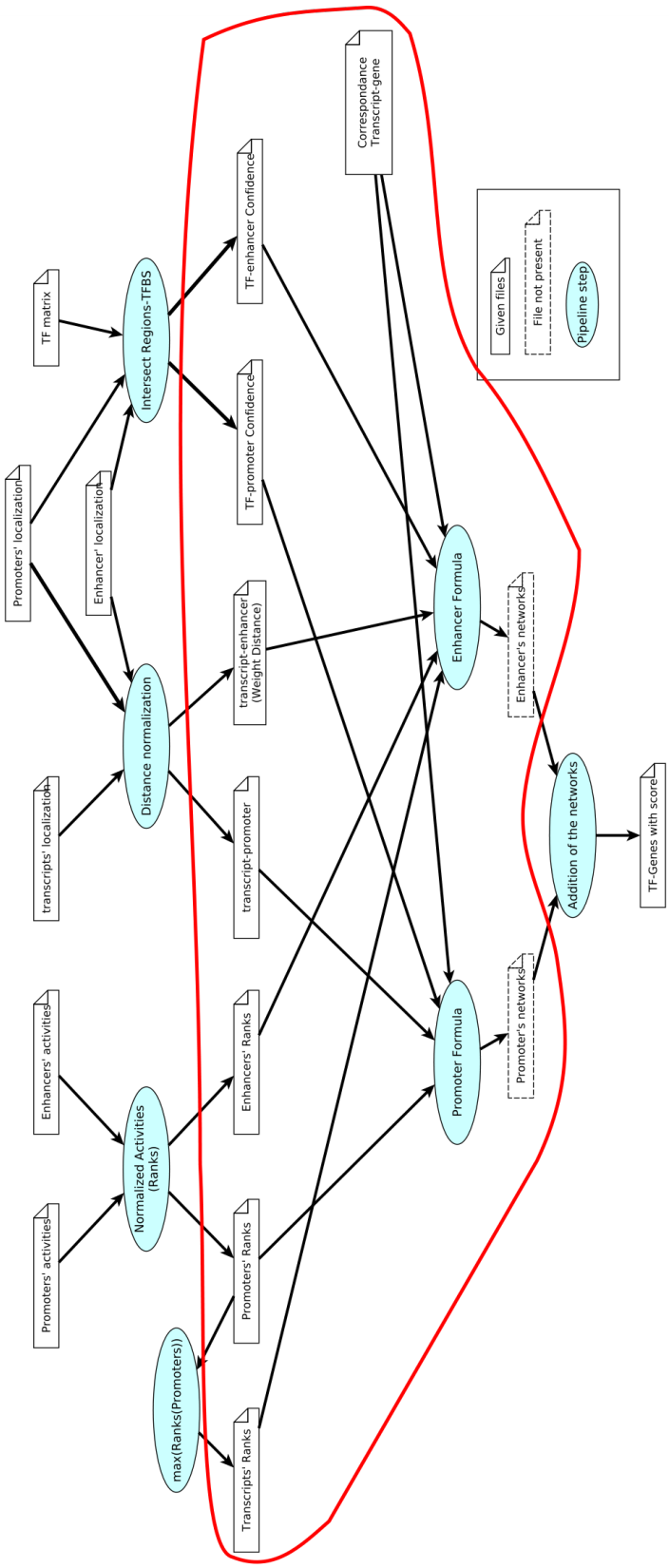


FIGURE 4.1 – *Regulatory Circuits* global workflow, circled in red are the files and steps of the workflow reused in this chapter: the steps are re-computed using SPARQL queries.

data (input) and pre-processed (intermediary) files, to recreate the published pipeline. TABLE 4.1 presents a review of the supplementary files in Regulatory Circuits, including the number of headers and comment lines, the entity names, and their format. Regulatory Circuits files also contained the computed networks, available on the download tab of their website under the Networks category. *FANTOM5_individual_networks.tar* contained 394 tissues-specific networks and *Network_compendium.zip* contained 32 high-level networks and 40 public ones. We did not use those final networks to construct our model.

On the 21 files present in TABLE 4.1, 14 were input files and seven were intermediary ones. The dataset was composed of text files of various size ranging from 184 to 124,358,159 lines and from 3 to 890 columns. This lead to large files which were complex to explore and made retrieving specific information difficult. For example the file *hg19.cage_peak_OK.txt* was just over 1.1GB.

These files had heterogeneous structures of headers and entities identifiers. 5 files had no header and one had 3 header lines. 3 of the files with headers were mis-formatted (*enhancer_expr.rank.txt*, *promoter_expr.rank.prec90.txt* and *transcript _expr.rank.prec90.txt*). They had an offset of 1 between the number of columns in the header versus in the data, which forced us to retrieve the data of the (n+1)th column to get the information related to the nth element. This contributed to the complexity of navigating those files. One file also had 800 comment lines above file header and one (*motif_defs.txt*) which contained only comments and non-formatted text.

To increase the difficulty of links retrieval between the files, the entity identifiers were not homogeneous across the dataset. For example, the promoter regions had an identifier sometimes following the pattern: *chr:start-end,strand* and some other times following: *p@chr:start..end,strand* (with chr being the chromosome on which is the region and $start$ and end are its chromosomal locations). Samples' names also differed across files headers. The most common denomination was the *libld* identifier based on $CNhs + nb$ where nb is a five digit integer (e.g.: *CNhs11051*), but in *hg19_permissive_enhancers_expression_rle_tpm.csv* the sample name were $cellType+ donor+nb : libld$ with $cellType$ either only the cell line or the cell line and the localization (example: *Adipocyte - breast donor1 : CNhs11051*). In *hg19.cage _peak _coord _robust.bed* this identifier were $tpm.Adipocyte\%20\%20breast\%2c\%20donor1.CNhs11051.11376-118A8$.

To identify which files were necessary to rebuild the regulation networks, we mapped the entities and files on the biological background (see CHAPTER 3) as shown in FIGURE 4.2.

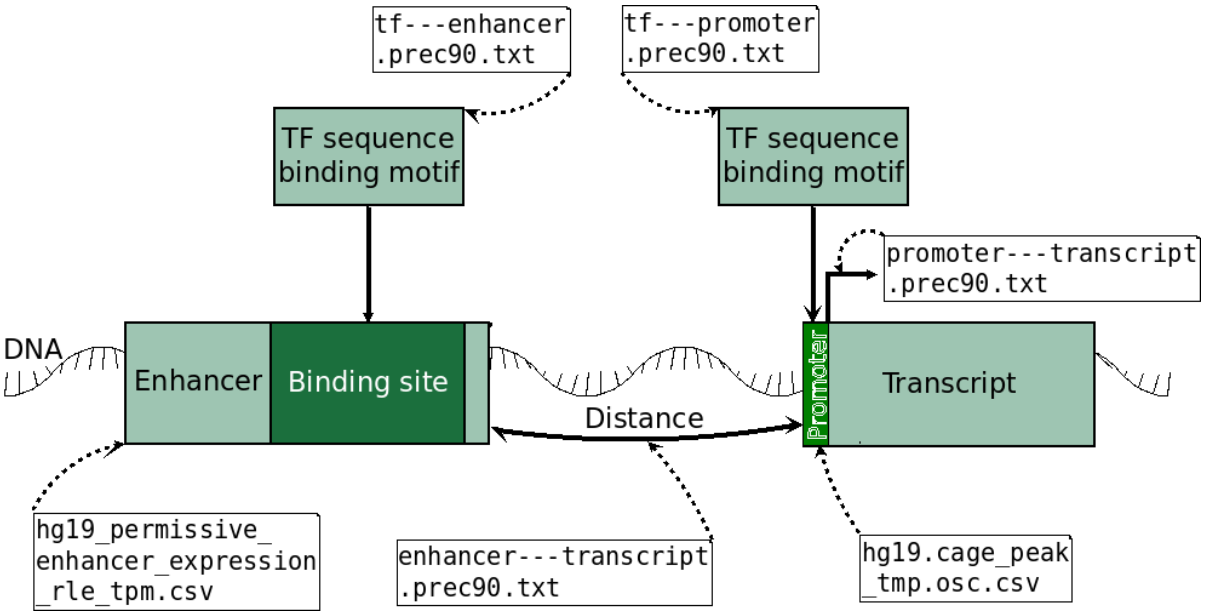


FIGURE 4.2 – Underlying biological background used to infer TF-gene relationships by the Regulatory circuits pipeline. TF can interact with enhancers or promoters binding sites. Regulatory Circuits original files containing the necessary entities and relations are also represented.

For the datatype values linked to entities, only elements that improved the recovery of entities by users were kept, such as: binding sites motif for transcription factors, ENSEMBL identifiers for genes and transcripts and DNA strands on which promoters are located.

4.2.2 Structuration

Once we identified all the files, entities and relations required to build TF-genes interactions, we structured their content. To do so, we first placed the entities in a graph as shown in FIGURE 4.3A. The only attributes given at this step were the expression levels of enhancers and promoters.

Second, we retrieved the relationships given in the Regulatory Circuits data. For those, we used the pre-computed distances between the transcripts and the regulatory

TABLE 4.1 – Regulatory Circuits files' review

type of file	file name	format	header lines	missformated header	comment lines	data lines	nb columns	Label of column(s) with ID	ID format	Entities	Source	content
Input data for network inference	*hg19_permissive_enhancers_expression_rle_tpm.csv*	csv (,)	1		0	43011	809	1	chr:start-end	enhancer	[1]	[a]
	permissive_enhancers.bed	bed12 (tab-delim)	1		0	43011	12	4	chr:start-end	enhancer	[1]	[b]
	robust_enhancers.bed	bed12 (tab-delim)	1		0	38554	12	4	chr:start-end	enhancer	[1]	[b]
	'hg19.cage_peak_tpm.osc.txt	tab-delim	3		893	184827	890	1	chr:start-end,strand	promoter	[2]	[a]
	hg19.cage_peak_coord_robust.bed	bed12 (tab-delim)	0		0	184827	12	4	chr:start-end,strand	promoter	[2]	[b]
	gene_coord.bed	bed6 (tab-delim)	0		0	19125	6	4	GENE_SYMBOL	gene	[3]	[b]
	gene_ids.txt	tab-delim	1		0	19125	3	1	ENSG00000000000	gene	[3]	[c]
								2	GENE_SYMBOL	gene		
								3	EntrezID	gene		
	mhc_genes.txt	tab-delim	1		0	184	1	1	GENE_SYMBOL	gene	[3]	[c]
	transcript_coord.bed	bed6 (tab-delim)	0		0	53449	6	4	GENE_SYMBOL-000	transcript	[3]	[b]
	transcript—gene.txt	tab-delim	1		0	53449	4	1	GENE_SYMBOL-000	transcript	[3]	[c]
								2	GENE_SYMBOL	gene		
								3	ENSG00000000000	transcript		
								4	ENST00000000000	gene		
	tss_coord.bed	bed6 (tab-delim)	0		0	53449	6	4	GENE_SYMBOL-000	gene	[3]	[b]
	motif_defs.txt	space-delim	0		1772	N/A	N/A	N/A	N/A		[4]	[g]
	motif_instances.bed	bed6 (tab-delim	0		0	124358159	6	4	TF_0	TF	[4]	[b]
	tf_motif_ids.txt	tab-delim	1		0	1792	3	1	TF	TF	[4]	[g]
								2	TF_0	TF		
Intermediary files	*enhancer_expr.rank.txt*	tab-delim	1	x	0	43011	809	1	e@chr:start..end	enhancer	[5]*	[d]
	enhancer—transcript.prec90.txt	tab-delim	1		0	950513	5	1	e@chr:start..end	enhancer	[5]*	[e]
								2	GENE_SYMBOL-000	transcript		
								5	GENE_SYMBOL	gene		
	promoter_expr.rank.prec90.txt	tab-delim	1	x	0	59126	809	1	p@chr:start..end,strand	promorer	[5]*	[d]
	promoter—transcript.prec90.txt	tab-delim	1		0	123440	4	1	p@chr:start..end,strand	promoter	[5]*	[e]
								2	GENE_SYMBOL-000	transcript		
								4	GENE_SYMBOL	gene		
	tf—enhancer.prec90.txt	tab-delim	1		0	524816	3	1	TF	TF	[5]*	[f]
								2	e@chr:start..end	enhancer		
	tf—promoter.prec90.txt	tab-delim	1		0	1169797	3	1	TF	TF	[5]*	[f]
								2	p@chr:start..end,strand	promoter		
	transcript_expr.rank.prec90.txt	tab-delim	1	x	0	43352	809	1	GENE_SYMBOL-000	trancript	[5]*	[d]

[1]http://enhancer.binf.ku.dk/Pre-defined_tracks.html, [2] http://fantom.gsc.riken.jp/5/datafiles/latest/extra/CAGE_peaks/, [3] Ensembl biomart, [4]Pouya Kheradpour (pouyak <a> mit.edu), [5]* Regulatory Circuits: auto produced, [a] Normalized activities,[b] Genomic coordinates,[c] Identifier, [d] Rank of normalized activities,[e] Distances, [f] Confidence score, [g] TF motifs

regions (enhancers and promoters), as well as the weight of the transcripts-enhancer distances. We also kept all the pre-processed confidence scores for the transcription factors / regions interactions. In the Regulatory Circuits article, the authors used a rank normalization of their expressions data in the final pipeline, so we used their intermediary files including these ranks for the enhancers and promoters expressions. We also kept the file including the rank for the transcript. All these interactions are described in FIGURE 4.3B.

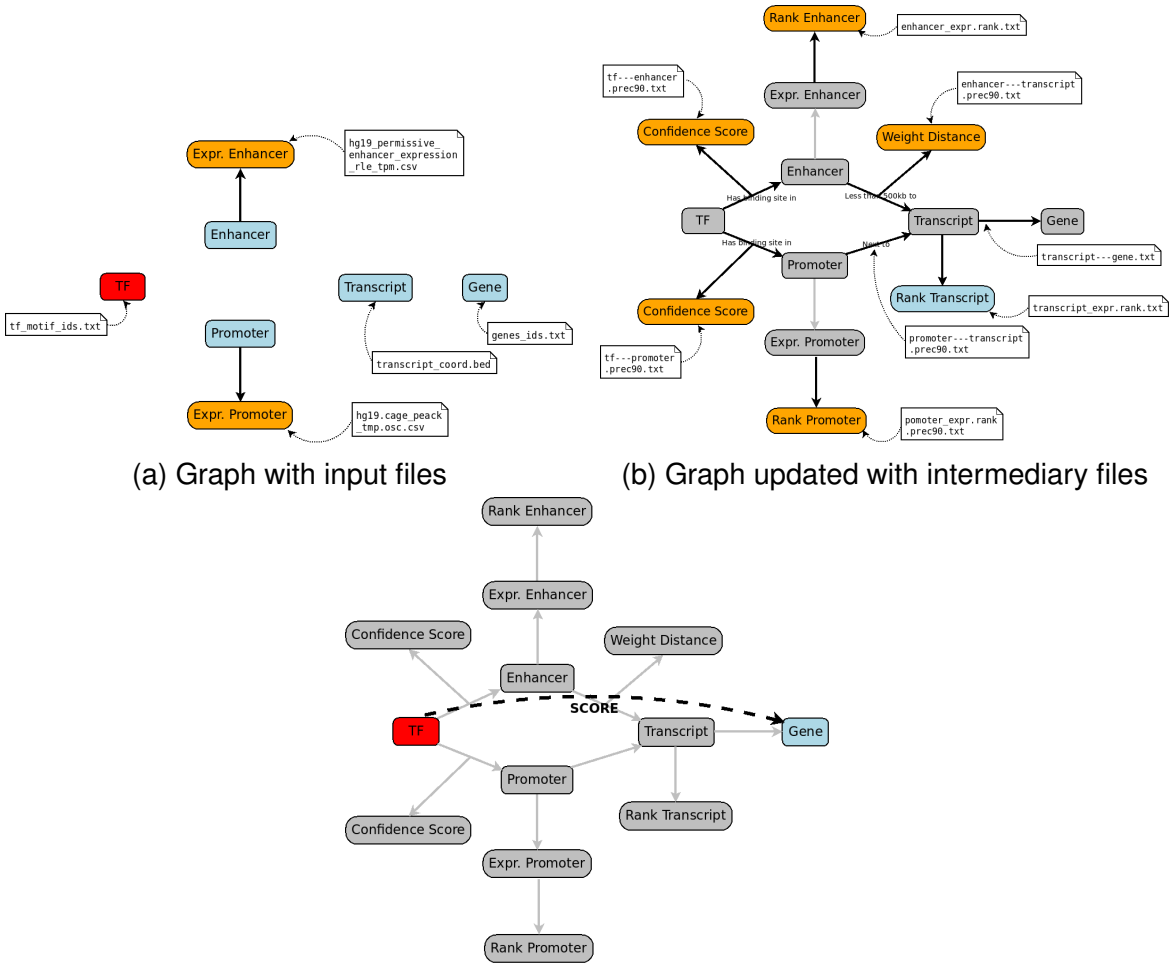
Third, the structure built from all these data and their interactions allowed us to easily retrieve the TF-gene relationships by navigating through the entities and their relations (FIGURE 4.3C).

4.2.3 Integration

Once the data had been structured, we integrated them so that they can be browsed and queried. To do so, we unified the identifiers and explicited the links between the entities.

We created a new set of rules to homogenize the entities identifiers across files ni order to facilitate integration. Regions identifiers were created using the following pattern: $r\_chrX\_start\_end$, r being the first letter of the region type (e for enhancer or p for promoter), X the chromosome number for the region and $start$ and end its chromosomal coordinates. For the expression, we chose to keep the *libld* identifier ($CNhs + nb$ with nb a five digit integer) of the tissues samples as name and added $Rank\_$ before this identifier for the ranks score of the same samples. Genes, transcripts and TF retained their original identifiers.

When a relation involved more than two entities or had some attributes, we used reification and represented the relation as an additional entity. The identifier for the reified relation was defined as $name1\_name2\_nb$ with *name1* the type of the first entity in the link, *name2* the second type and *nb* a unique integer. The reified relation was then associated to the entities and attributes using regular binary relations (for example in FIGURE 4.3B, notice that the relation from a TF to a Promoter had a confidence score ($confidence$); this ternary relation was represented by the *tf_promoter* entity in the RDF model in FIGURE 4.4 which associated a TF, a Promoter and a confidence). We created a RDF graph of the dataset using Regulatory Circuits data and new entities for



(a) Graph with input files

(b) Graph updated with intermediary files

(c) Final structure

FIGURE 4.3 – The three steps of data structuring, with identification of the files containing the needed information. In (a) identification of input files. In this step we mostly import entities (genes, TF, regulatory regions). The only imported relation is the expression levels of both types of regulatory regions.
In (b) we added all the information from the intermediary files: interactions between the different elements and scores based on those relations. We also added pre-processed scores on the expression levels, called Ranks.
In (c) we can see that the TF-gene relations were obtained by following the links between entities and that these relations could be weighted using the score from step (b).

representing reified relationships, as shown in FIGURE 4.4. The description of entities classes, numbers and attributes in each node can be found in TABLE 4.2.

Once the files had been cleaned and augmented with reified relationship, we organised them as coma separated files. We integrated these data using AskOmics, using the formatted files as entries. AskOmics auto-generated the RDF files associated with the data and use its specific prefix, as can be seen in queries below. Due to the file size limitations, we had to separate some of the files in smaller ones resulting in a total of forty-four integrated files.

We then deployed them as a SPARQL endpoint using OpenLink Virtuoso engine. [2] As shown in TABLE 4.3, we integrated ten classes representing more than three hundred million triples.

The description of the dataset population can be seen in FIGURE 4.4b and Tables 4.2 and 4.3: over three million entities, separated in ten classes, each with several attributes.

4.2.4 Queries

After integrating all the data, we could query the dataset in order to retrieve the TF-gene relationships for each cell type or tissue. According to Regulatory Circuits there are two ways of getting the transcription factor and gene relationship: using either type of regulatory regions (enhancers or promoters). The first step consisted in computing all the potential TF-gene relations.

The first query used the promoter as the binding region for the TF: starting from the TF, and continuing by the promoter, the transcript and then the gene. To confirm the existence of this relation, we needed to verify that the TF confidence score and the promoter expression rank were both different from 0.

```
PREFIX user: <http://www.semanticweb.org/user/ontologies/2018/1#>
PREFIX askomics: <http://www.semanticweb.org/askomics/ontologies/2018/1#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
SELECT DISTINCT ?tf1 ?gene1
WHERE {
  ?tf1 rdf:type user:tf.
```

2. The RDF dataset can be retrieved from *https:// regulatorycircuits-rdf.genouest.org/ dump/* and the SPARQL endpoint is accessible at *https:// regulatorycircuits-rdf.genouest.org/ sparql*

TABLE 4.2 – Data population. The first three lines correspond to the nodes in red in FIGURE 4.4, the next two correspond to the blue nodes and the last five to the pink ones.

Class ***(nb of entities)***	**Attributes**
Gene *(19 125)*	ID_ENSEMBL
Transcript *(53 549)*	ID_ENSEMBL Rank for 808 pop.
TF *(691)*	motif description
Promoter *(184 828)*	strand Expression for 889 pop. Rank for 808 pop.
Enhancer *(43 011)*	Expression for 808 pop. Rank for 808 pop.
tf_promoter *(1 169 797)*	confidence inclu@tf in@promoter
tf_enhancer *(524 816)*	confidence inclu@tf in@enhancer
transcript_gene *(53 449)*	isgene@gene istranscript@transcript
promoter_transcript *(123 441)*	distance nextto@transcript nextto@promoter
enhancer_transcript *(950 514)*	distance weight nextto@transcript nextto@promoter

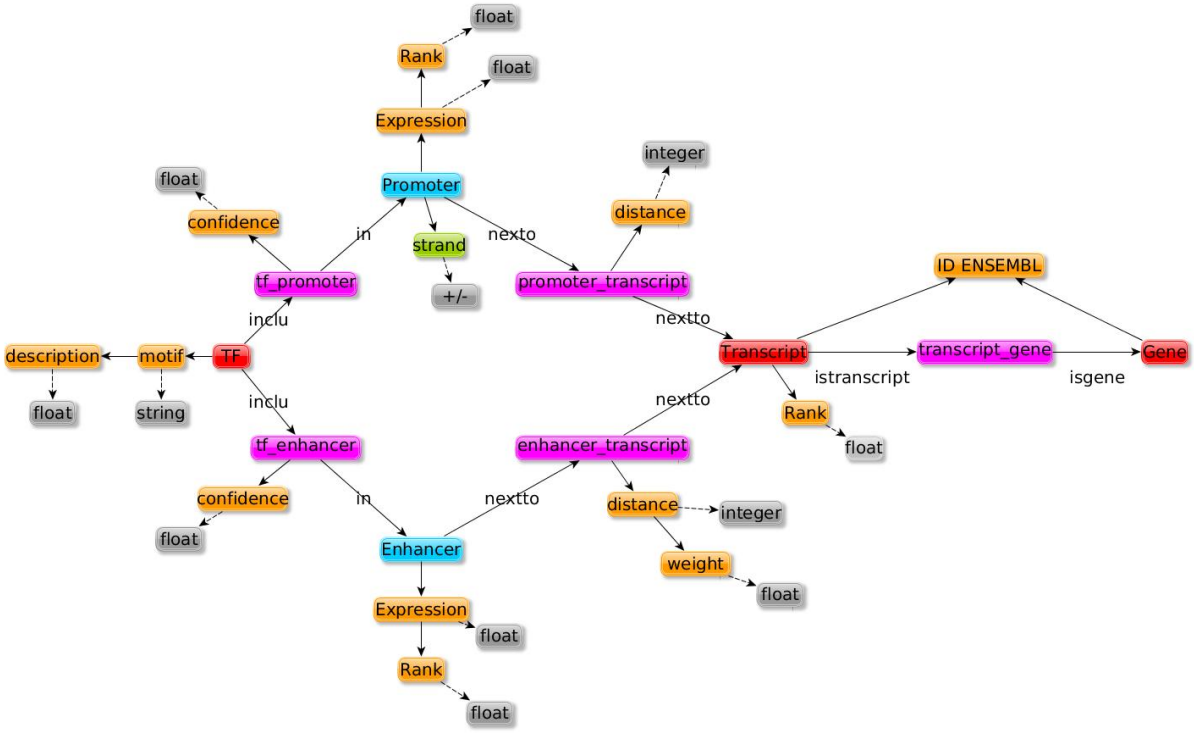


FIGURE 4.4 – Data structure after integration. Nodes in red are gene entities, nodes in blue are regulatory region entities and nodes in pink are reified relations. Genomic localization attributes are indicated in green, other attributes are in orange and attributes type are in gray. Details about the number of entities for each nodes can be found in TABLE 4.2.

```
  ?tf_promoter1 rdf:type user:tf_promoter.
  ?tf_promoter1 askomics:confidence ?confidence1.
  FILTER ( ?confidence1 > 0 ).
  ?promoter1 rdf:type user:promoter.
  ?promoter1 askomics:Rank_CNhs12017 ?Rank_CNhs12017P.
  FILTER ( ?Rank_CNhs12017P > 0 ).
  ?promoter_transcript1 rdf:type user:promoter_transcript.
  ?transcript1 rdf:type user:transcript.
  ?transcript_gene1 rdf:type user:transcript_gene.
  ?gene1 rdf:type user:gene.
  ?tf_promoter1 askomics:inclu ?tf1.
  ?tf_promoter1 askomics:in ?promoter1.
  ?promoter_transcript1 askomics:nextto ?promoter1.
  ?promoter_transcript1 askomics:nextto ?transcript1.
  ?transcript_gene1 askomics:istranscript ?transcript1.
  ?transcript_gene1 askomics:isgene ?gene1.
}
ORDER BY ?tf1 ?gene1
```

TABLE 4.3 – Integrated data

	Number of elements
Triples	335 429 988
Entities	3 226 341
Classes	10
Datasets	53

The second query was similar but used the enhancer instead of the promoter: we started from the TF, and proceeded following the enhancer, the transcript and then the gene, making sure that all score component were superior to 0.

```
PREFIX user: <http://www.semanticweb.org/user/ontologies/2018/1#>
PREFIX askomics: <http://www.semanticweb.org/askomics/ontologies/2018/1#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
SELECT DISTINCT ?tf1 ?gene1
WHERE {
  ?tf1 rdf:type user:tf.
  ?tf_enhancer1 rdf:type user:tf_enhancer.
  ?tf_enhancer1 askomics:confidence ?confidence1.
 FILTER ( ?confidence1 > 0 ).
  ?enhancer1 rdf:type user:enhancer.
  ?enhancer1 askomics:Rank_CNhs12017 ?Rank_CNhs12017E.
  FILTER ( ?Rank_CNhs12017E > 0 ).
  ?enhancer_transcript1 rdf:type user:enhancer_transcript.
  ?enhancer_transcript1 askomics:weight ?weight1.
  FILTER ( ?weight1 > 0 ).
  ?transcript1 rdf:type user:transcript.
  ?transcript1 askomics:CNhs12017 ?CNhs12017T.
  FILTER ( ?CNhs12017T > 0 ).
  ?transcript_gene1 rdf:type user:transcript_gene.
  ?gene1 rdf:type user:gene.
  ?tf_enhancer1 askomics:inclu ?tf1.
  ?tf_enhancer1 askomics:in ?enhancer1.
  ?enhancer_transcript1 askomics:nextto ?enhancer1.
  ?enhancer_transcript1 askomics:nextto ?transcript1.
  ?transcript_gene1 askomics:istranscript ?transcript1.
  ?transcript_gene1 askomics:isgene ?gene1.
}
ORDER BY ?tf1 ?gene1
```

The two first queries were designed using AskOmics, in which the SPARQL code for the queries are automatically generated using the intuitive graphical interface (see

FIGURE 4.5), following the relation between the entities and allow to retrieve attributes along the way.

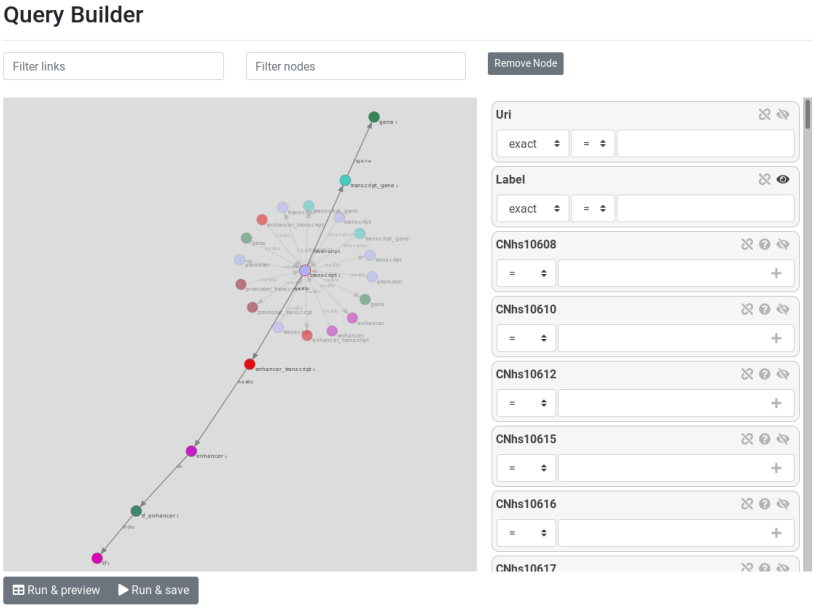


FIGURE 4.5 – Graphical based query for the relation between a TF and a Gene, using enhancer. The block on the left is the query builder and the list on the right is the definition of the attributes we might want to retrieve or put under conditions.

In the final Regulatory Circuits network, all the TF-gene relations were qualified by a score (cf. FIGURE 4.3C). The score ($w_{ij}(S)$) between the transcript (j) and the TF (i) in the sample S is given based on the distance weight(d_{ik}) between the transcript and the regulatory region (k), the confidence score of the binding site of the TF in the region (c_{ik}), the normalisation of the expression of the region (x_k) and the normalisation of the expression of the transcript (y_j).

$$w_{ij}(S) = c_{ik} \times d_{kj} \times \sqrt{x_k(S) \times y_j(S)}$$

This score was the maximum of all the TF-gene relations scores obtained through either promoters (bottom part of FIGURE 4.3C) or enhancers (top part of FIGURE 4.3C). The intermediate score through a promoter was $Confidence\_Score \times Rank\_promoter$. As for the promoter the distance weight is normalized to 1 and the rank of the transcript is the same as the rank of the promoter.

$$w_{ij}(S) = c_{ik} \times x_k(S)$$

The intermediate score through an enhancer was $Confidence\_Score \times Weight\_Distance \times \sqrt{(Rank\_transcript \times Rank\_enhancer)}$ where $Rank\_transcript$ (y_j) is the max of the transcript promoters ranks.

$$w_{ij}(S) = c_{ik} \times d_{kj} \times \sqrt{x_k(S) \times y_j(S)}$$

```
PREFIX user: <http://www.semanticweb.org/user/ontologies/2018/1#>
PREFIX askomics: <http://www.semanticweb.org/askomics/ontologies/2018/1#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
SELECT DISTINCT ?tf1 ?gene1 (max(xsd:float(?confidence1) * xsd:float(?confidence1) * xsd:float(?Rank_CNhs12017P) *
    xsd:float(?Rank_CNhs12017P)) AS ?weightP)
WHERE {
  ?tf1 rdf:type user:tf.
  ?tf_promoter1 rdf:type user:tf_promoter.
  ?tf_promoter1 askomics:confidence ?confidence1.
  FILTER ( ?confidence1 > 0 ).
  ?promoter1 rdf:type user:promoter.
  ?promoter1 askomics:Rank_CNhs12017 ?Rank_CNhs12017P.
  FILTER ( ?Rank_CNhs12017P > 0 ).
  ?promoter_transcript1 rdf:type user:promoter_transcript.
  ?transcript1 rdf:type user:transcript.
  ?transcript_gene1 rdf:type user:transcript_gene.
  ?gene1 rdf:type user:gene.
  ?tf_promoter1 askomics:inclu ?tf1.
  ?tf_promoter1 askomics:in ?promoter1.
  ?promoter_transcript1 askomics:nextto ?promoter1.
  ?promoter_transcript1 askomics:nextto ?transcript1.
  ?transcript_gene1 askomics:istranscript ?transcript1.
  ?transcript_gene1 askomics:isgene ?gene1.
}
GROUP BY ?tf1 ?gene1
ORDER BY ?tf1 ?gene1
```

With our structured data we could extend our queries to compute the intermediate promoter and enhancer-related scores. SPARQL queries do not support square root, but could easily be devised to compute the square of the previously presented scores. For enhancers, although we could have written queries that compute $Rank\_transcript$ on the fly (and recompute it for each transcript every time a promoter is considered), we took advantage of the intermediary files where $Rank\_transcript$ values were already provided, and added these pre-computed $Rank\_transcript$ values to our RDF model.

The resulting query for computing the score could then use directly the $Confidence\_Score \times Weight\_Distance \times \sqrt{(Rank\_transcript \times Rank\_enhancer)}$ formula.

```
PREFIX user: <http://www.semanticweb.org/user/ontologies/2018/1#>
PREFIX askomics: <http://www.semanticweb.org/askomics/ontologies/2018/1#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
SELECT DISTINCT ?tf1  ?gene1 (max(xsd:float(?confidence1) * xsd:float(?confidence1) * xsd:float(?weight1)*
  xsd:float(?weight1) * xsd:float(?CNhs12017T) * xsd:float(?Rank_CNhs12017E) ) AS ?weightE)
WHERE {
  ?tf1 rdf:type user:tf.
  ?tf_enhancer1 rdf:type user:tf_enhancer.
  ?tf_enhancer1 askomics:confidence ?confidence1.
  FILTER ( ?confidence1 > 0 ).
  ?enhancer1 rdf:type user:enhancer.
  ?enhancer1 askomics:Rank_CNhs12017 ?Rank_CNhs12017E.
  FILTER ( ?Rank_CNhs12017E > 0 ).
  ?enhancer_transcript1 rdf:type user:enhancer_transcript.
  ?enhancer_transcript1 askomics:weight ?weight1.
  FILTER ( ?weight1 > 0 ).
  ?transcript1 rdf:type user:transcript.
  ?transcript1 askomics:CNhs12017 ?CNhs12017T.
  FILTER ( ?CNhs12017T > 0 ).
  ?transcript_gene1 rdf:type user:transcript_gene.
  ?gene1 rdf:type user:gene.
  ?tf_enhancer1 askomics:inclu ?tf1.
  ?tf_enhancer1 askomics:in ?enhancer1.
  ?enhancer_transcript1 askomics:nextto ?enhancer1.
  ?enhancer_transcript1 askomics:nextto ?transcript1.
  ?transcript_gene1 askomics:istranscript ?transcript1.
  ?transcript_gene1 askomics:isgene ?gene1.
}
GROUP BY ?tf1 ?gene1
ORDER BY ?tf1 ?gene1
```

We then computed the score for TF-gene relations as the square root of the maximum of both the promoter and enhancer queries. Overall, the complete Regulatory Circuits pipeline producing both TF-gene relations and their associated scores could be performed by 2 SPARQL queries. Theses queries were rather simple and involved 7 kinds of entities and 6 relations.

All queries in this section were based on the CNhs12017 sample of Regulatory Circuits and can be extended to other tissues by changing the sample name in the queries. The full list of tissue samples and their descriptions is given in the supplementary data file *nmeth.3799-S2.xlsx* from Regulatory Circuits [3]. A sub-list of samples

3. Link to nmeth.3799-S2.xlsx

names is given in Table 4.4 in the following section.

4.2.5 Performances

Performance-wise, TABLE 4.4 shows that all queries times ranged from 4.49 seconds for the fastest and 537.32 seconds (9 minutes) for the longest. FIGURE 4.4 shows the re-partition of the execution times. On the 3.232 queries (4 queries for each 808 samples) only 124 had an execution time over 90 seconds. Each of the 4 queries have been performed on the 808 different samples of the dataset. This have been automated by using the python SPARQLwrapper library and feeding it the list of all different sample names.

TABLE 4.4 – Queries' execution time (in seconds) for some of the 808 samples. They were run on the SPARQL end-point *https:// regulatorycircuits-rdf.genouest.org/ sparql*. The means are over the 808 samples.

	Queries for TF-relation based on: (in seconds)			
Sample name	**Promoters all $>$ 0**	**Enhancers all $>$ 0**	**Promoters & Score**	**Enhancers & Score**
CNhs12017	19.310	16.359	31.319	20.515
CNhs13465	18.062	50.649	28.630	70.771
CNhs10629	23.631	20.755	37.505	27.434
CNhs11750	16.519	5.437	26.650	6.915
CNhs13195	16.138	26.339	28.377	38.867
CNhs13492	18.159	44.711	25.378	66.119
CNhs11771	22.666	13.026	30.768	16.451
CNhs12347	17.099	9.029	28.260	11.861
CNhs11047	21.361	35.926	35.727	49.122
CNhs12075	17.775	10.570	26.353	12.648
CNhs13099	16.105	9.275	24.457	12.225
CNhs12569	20.792	15.234	32.980	19.403
CNhs10636	28.768	51.718	42.956	75.017
CNhs11869	19.686	14.204	29.645	19.136
...	...	...	...	...
Fastest	**12.064**	**4.487**	**18.054**	**4.734**
Slowest	**148.232**	**329.655**	**217.806**	**537.319**
Mean	**27.189**	**32.060**	**42.500**	**43.798**

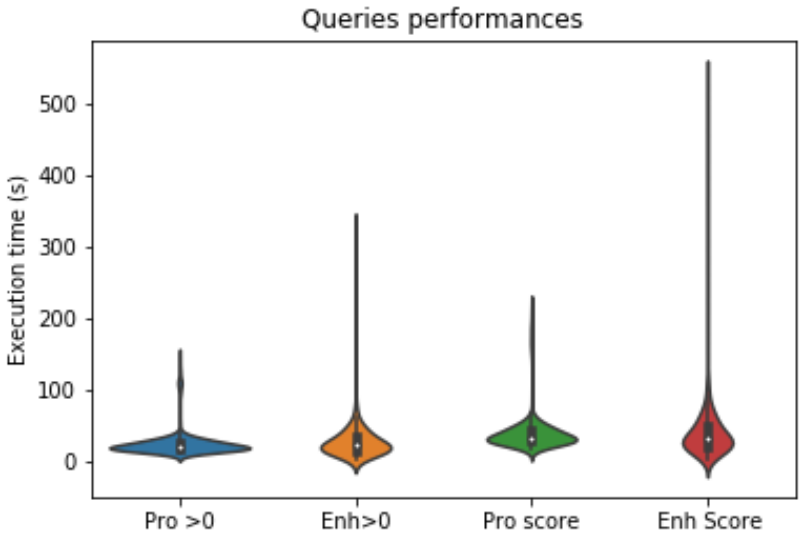


FIGURE 4.6 – Visual representation of queries' execution time, in second, for all the 808 samples.

We chose to have two distinct queries to retrieve yhe TF-gene relations scores, and to process their results to keep the maximal score instead of an unique query which would result in longer execution time.

4.3 Discussion

Our approach consisted in structuring the data and results of a systems biology study as a RDF dataset. Our experience was that reusing the 21 raw and intermediary files from Regulatory Circuits required an in-depth analysis of their structure and of the documentation. We produced a RDF model (FIGURE 4.4) of Regulatory Circuits that provides a unified access to their networks which are currently spread in 394 cell types and tissue-specific files, statically grouped into 32 high-level regulatory networks. This RDF model saves future users from having to manually reproduce the integration effort. Our results showed that once the relations and ranks had been pre-computed, the Regulatory Circuits analysis pipeline could be formalized as two SPARQL queries. We argue that this unified RDF dataset makes querying and reuse in other studies easier.

Even if the structure of our RDF model (FIGURE 4.4) is fairly simple, the Regulatory Circuits dataset is rather large (more than 300 millions triples, cf. TABLE 4.3). Despite the size, SPARQL querying performances were of a few seconds (TABLE 4.4).

The Regulatory Circuits pipeline relies on raw data as well as external resources such as Ensembl that are regularly updated. To accomodate these updates, the original Regulatory Circuit data structure requires to update some raw files, regenerate the intermediary files that depend on them and run the pipeline. With our approach, these third-party updates can easily be propagated to our RDF model by running the SPARQL queries.

The RDF version of Regulatory Circuits allows a fine-grained exploration of the relations between entities (transcription factors, enhancers, promoters, transcripts and genes) involved in regulation mechanisms. For example, it allows to differentiate the relations involving enhancers from the ones involving promoters (e.g. for taking into account that promoters relations are more reliable). Similarly, it allows to differentiate between the binding motifs of a single transcription factor or to consider transcription factors from a specific family that usually share similar binding site motifs.

The RDF version of Regulatory Circuits can also be extended with user-specific data, which increases flexibility. For example, if users have expression data of additional tissues, a new set of regulatory regions or binding data for an undescribed transcription factor, they can update the current model to add their new data. Depending on the type of data it may require pre-processing, to fit with Regulatory Circuits current dataset. Users can also import new data not present in the current data structure by following the rules described in SECTION 4.2. This will require to extend the RDF graph (FIGURE 4.4), which is straightforward in RDF.

Following the Linked Data approach, we used the Ensembl identifiers for genes and transcripts. Federated SPARQL queries can then be used to combine information for Regulatory Circuits with information from Ensembl (e.g. variants, associations with diseases, or annotations).

Our approach is rather generic and should be straightforward to other studies for which the analysis pipeline follows relations and performs simple arithmetic functions, such as parts of the ENCODE or Roadmap Epigenomics databases. More in-depth analyses (e.g. statistical) are beyond SPARQL expressivity and should be addressed by dedicated pre or post-processing.

4.4 Conclusion

Life Science current standardization and integration efforts increasingly rely on Semantic Web technologies. They are currently directed towards reference data and knowledge bases. We hypothesized that applying the same approach to original studies would improve the results reproducibility, their maintenance and their reuse for advancing other studies. We considered the Regulatory Circuits case-study. We surveyed the 394 original data files and proposed an unified RDF data model. We showed that the Regulatory Circuits analysis pipeline can be formalized as two SPARQL queries and that the performances were acceptable. Overall, this unified RDF dataset makes querying and reuse in other studies easier.

CHAPITRE 5

WORKFLOW AND INTERMEDIARY RESULTS AS GRAPH-BASED QUERY AND MODEL

This chapter follows the work of the previous chapter (CHAPTER 4) and expends it, leading to scaling of the previous method to not only recover sample-specific networks but tissue-specific ones. This chapter is the body of an article in collaboration with: Fabrice Chatonnet, Xavier Garnier, Thierry Fest, Anne Siegel, Catherine Faron and Olivier Dameron, which is currently ready for submission.

We present in SECTION 5.1 the global approach of this chapters. In SECTION 5.2 we describe: (1) the design principle and the organisation of the developed resource, (2) the data and metadata used and integrated, (3) the queries to recover sample-specific TF-gene regulations, (4) the queries for tissue-specific relations. We also present (6) the overall data-set of the resource and (6) some examples of biologically relevant queries we can perform on the resource. In SECTION 5.3 we discuss the perspectives and the limitations of this resource and approach.

5.1 Introduction

In the previous Chapter [LOUARN et al., 2019], we provided a data structure enabling to integrate the source biological data of the *Regulatory Circuits* project in a RDF triplestore of 3,226,341 entities and 335,429,988 relations. As an application case-study, we evidenced that TF-gene interaction networks for each cell sample could be generated on-the-fly with two SPARQL queries.

Here, we elaborate upon this strategy to generate a public RDF resource which

contains not only the *Regulatory Circuits* source biological data, linked to standard LOD resources, but also the results of the analysis pipeline at the sample and tissue-specific layers. This extension from the previous chapter is presented in FIGURE 5.1, we do use the same entree files as before and re-compute the same first two steps, but after computing the promoters and enhancers networks we combine them to compute the final step of the original workflow: tissues-specific networks.

The expected benefits are threefold. First, instead of only having access to tissue-specific regulatory networks, biologists will also be able to query this resource from different perspectives. For example, they may be interested in comparing the targets of a given TF in different tissue-specific networks, or in determining how the TFs regulating a given gene vary among networks. Second, biologists will be able to define new tissue-specific regulatory networks based on the 808 samples from *Regulatory Circuits*. This encompasses both specializing a network by selecting a subset of the samples it is based on, or generalizing a network by adding other samples. Third, biologists will be able to combine the data from *Regulatory Circuits* with their specific samples. Altogether, this resource aims at providing a more complete, more flexible and reusable rendition of the *Regulatory Circuits* dataset.

5.2 Approach

Our first contribution is motivated by the important drawback of using on-the-fly SPARQL queries to compute TF-gene interaction networks. Each sample-specific network contains some values such as ranks that depend on values from the other networks, so that the 808 sample-specific networks have to be computed simultaneously. Therefore, exploring, comparing and integrating information about a subfamily of TF-gene interactions requires to compute the full set of networks, which necessitates large-scale computation resources. In order to save the users the trouble (and time, and CPU) to perform these queries, we execute them once (11.2 days CPU times) and integrate the final 808 sample-specific networks in our resource triplestore.

Our second contribution focuses on computing the tissue-specific networks by aggregating the corresponding sample-specific networks. We show that this is equivalent to a union SPARQL query on the source biological data and the 808 sample-specific networks from our first contribution. Even if these sample-specific networks were consi-

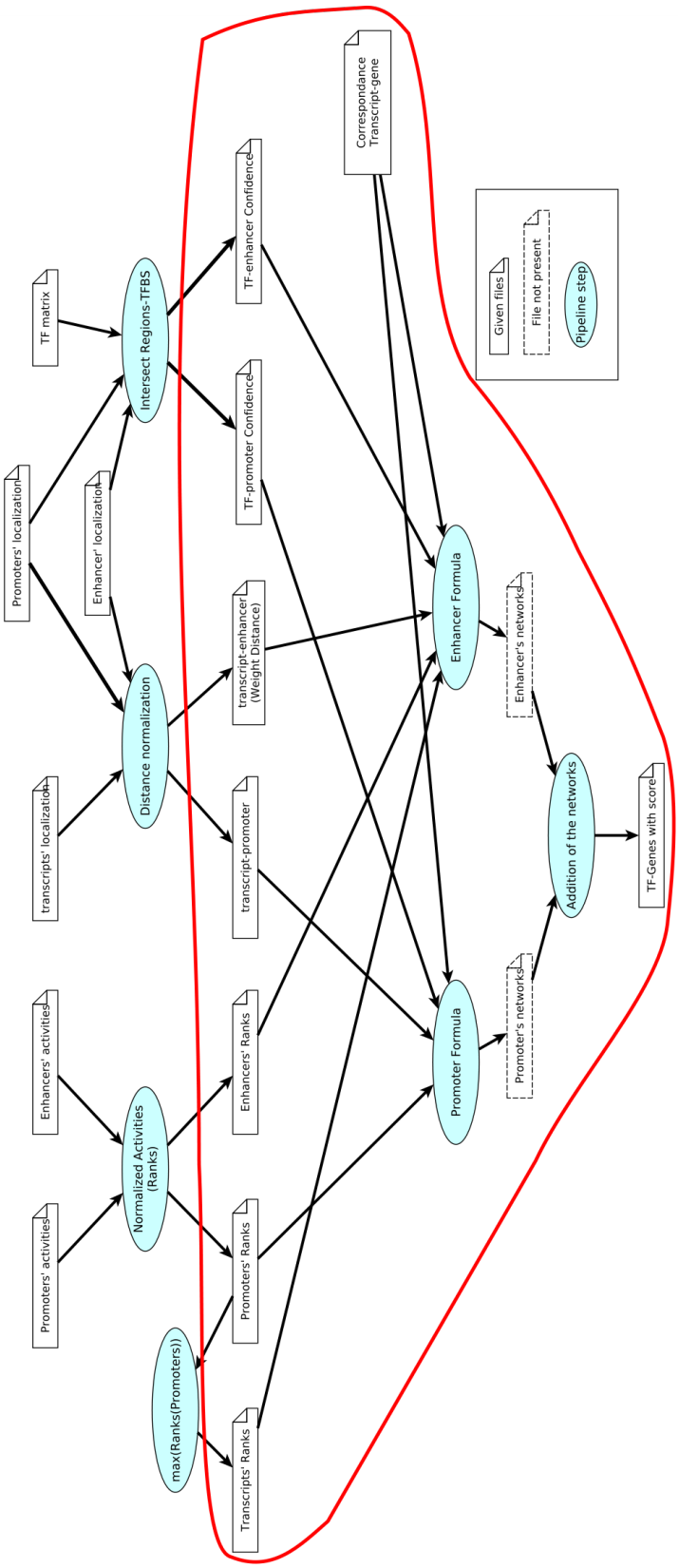


FIGURE 5.1 – *Regulatory Circuits* global workflow, circled in red are the files and steps of the workflow reused or re-computed in this chapter.

dered as intermediary (and unpublished) results in the original *Regulatory Circuits* pipeline, we recognize that they are crucial for computing tissue-specific networks. We compute the 394 tissue-specific TF-gene interaction networks, which are also included in the RDF triplestore to facilitate their reusability by users in global analyses.

Our third contribution elaborates on the capability to query only some portion of the *Regulatory Circuits* dataset relevant for the user. It consists in integrating metadata such as the descriptions of the samples (which types of cells they were measured in, information about the donor, etc.) and of the tissues (the organ they refer to, the studied pathology and of course the samples they were composed of), which were also provided as tabulated files by *Regulatory Circuits*. When applicable, we also include references to other knowledge bases from the Linked Open Data initiative such as gene identifiers from Ensembl, protein identifiers from Uniprot, cell types and anatomical structures from the Uberon ontology. This explicit representation of these metadata about the samples and the tissues as well as the relations to external knowledge bases can be queried by the users for identifying the portions of the dataset they are interested in. The result can then be combined with our second contribution to support flexibility.

Our fourth contribution is motivated by avoiding performance issues when integrating and querying large datasets. It consists in proposing a modular organization into named graphs for the original biological data from *Regulatory Circuits*, the 808 sample-specific and the 394 tissue-specific regulatory networks, as well as for the metadata.

We called *Linked Extended Regulatory Circuits (LERC)* our RDF representation of the *Regulatory Circuits*. Overall, our dataset consists in (i) descriptions of biological and experimental data and linked to the references databases, (ii) annotations about TF-gene interactions at the sample level for 808 samples, (iii) annotations about TF-gene interactions at the tissue level for 394 tissues, (iv) metadata connecting the named graphs of the three previous points. It contains 2,145,789,028 triples and required 28.6 days CPU to be generated. A Virtuoso endpoint is available at https://regulatorycircuits-lod.genouest.org/. The integration scheme to construct it is applicable to any similar dataset produced in other project.

5.3 Results

First, we describe our design principles and our modular organization into RDF named graphs. Second, we describe the biological data from *Regulatory Circuits* and the metadata associated to the samples and the tissues. Third, we describe the sample-specific graphs as well as the SPARQL queries for computing the weights associated to TF-genes relations and integrating them in the graphs. Fourth, we describe the tissue-specific graphs as well as the SPARQL queries for computing the weights and scores associated to TF-genes relations based on the values computed for the samples, and integrating them in the graphs. Eventually, we show how our flexible architecture supports biologically-relevant SPARQL queries that were not possible with our previous representation of *Regulatory Circuits*'s final results in RDF.

5.3.1 Design principles and modular organization

As shown in FIGURE 5.2, our RDF dataset encompasses a total of 1,205 graphs of five types: 1 source biological data graph (in blue), 1 experimental data graph (in purple), 808 sample-specific graphs (in green), 394 tissue-specific graphs (in orange) and 1 metadata graph (in grey). The RDF models for each type of graphs is described in the next sections. Their main characteristics are as follows:

— The source biological data graph representing the biological data of the *Regulatory Circuits* and FANTOM5 projects was already published in [LOUARN et al., 2019].

— The experimental context graph contains all the information about samples (organ, cell type and patient-related information) and tissues (the samples they are composed of).

— Each sample-specific RDF graph provides the weights of the TF-gene interactions associated with the considered sample.

— Each tissue-specific graph provides the weights and scores of the TF-gene interactions associated with the considered tissue-specific regulatory networks, which is an aggregation of biologically related individual samples.

— The metadata graph contains all the information about the other graphs including their VoID descriptions, as well as the associations of the samples and tissues with their respective graph.

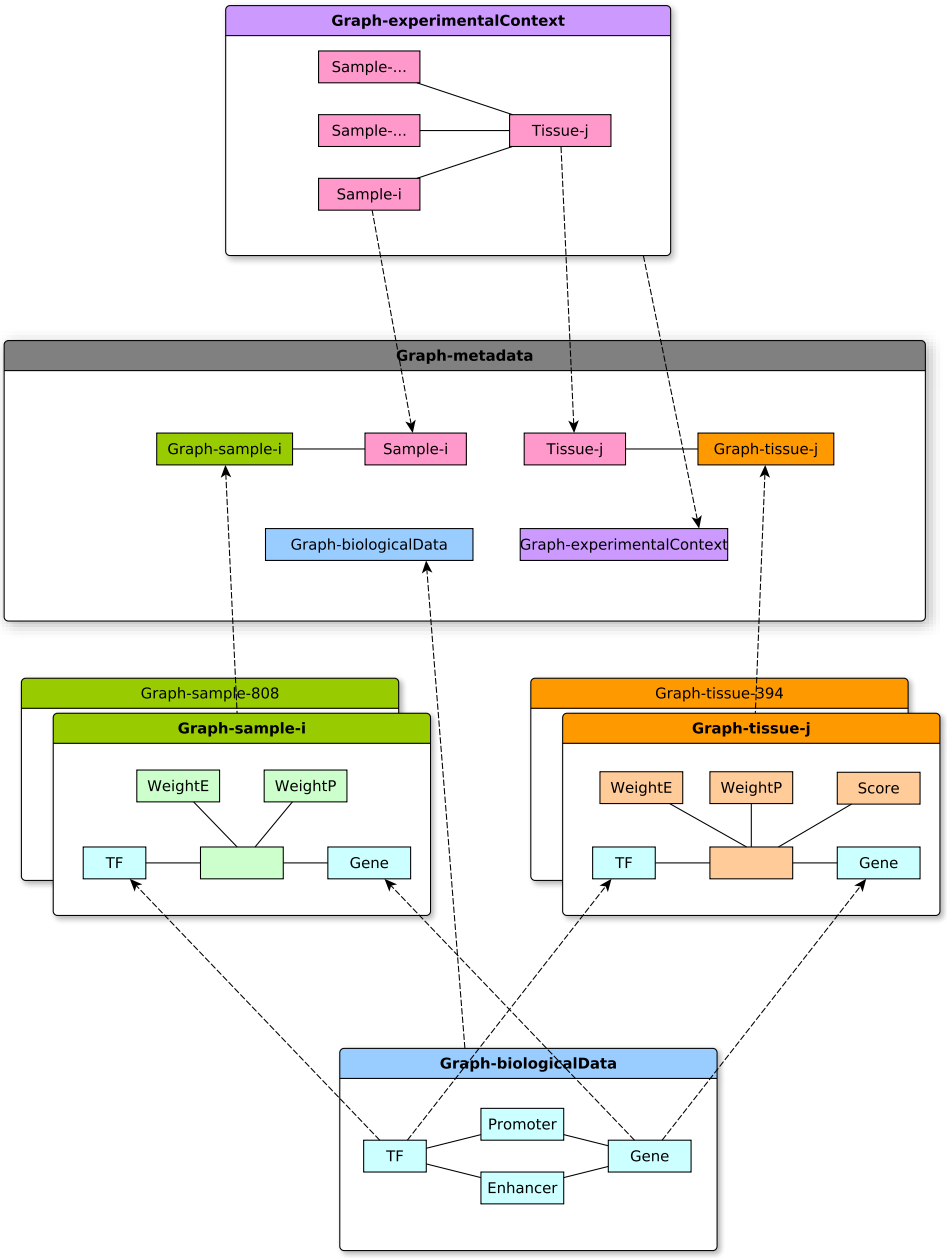


FIGURE 5.2 – Modular organization of the RDF dataset into 1,205 named graphs. The named graphs are the labeled boxes. The plain boxes and straight edges represent a simplified view of the main classes of each graph and their relations. The dotted arrows represent how the graphs are connected through entity matching, i.e. the URIs of entities primarily described in the graph at the origin of the edge are reused in the graph at the end of the edge.

5.3.2 Biological and experimental data from *Regulatory Circuits* and metadata

The graph of biological data (FIGURE 5.3) contains descriptions of the biological entities that are independent from the experiments. As detailed in [LOUARN et al., 2019] and depicted in FIGURE 5.3, it is based on five main biological entities: three related to genes or proteins (gene, transcript, TF) and two related to chromosomal regulation regions (promoter, enhancer), and on five reified relations.

The identifiers of genes (19,125 instances of the class $Gene$), transcripts (53,549 instances of the class $Transcript$) and transcription factors (691 instances of the class TF) are constructed by using the names provided by the *Regulatory Circuits* datasets (HGNC reference identifiers for TF and $Gene$, Ensembl transcript names for $Transcript$). These identifiers are linked to the external databases Uniprot and Ensembl identifiers as follows. Genes are associated to the Uniprot identifier of their reviewed proteins; in case of several proteins being reviewed for a gene, the longest one is selected. Both genes and transcripts are associated to their Ensembl identifiers as already available in *Regulatory Circuits* datasets.

There are two classes of regulatory regions: $Promoter$ (184,828 entities) and $Enhancer$ (43,011 entities).

The dataset comprises five types of reified relations, two between TFs and regulatory regions weighed by the $confidence$ of transcription factor binding site in the region (1,169,797 entities for $TF\_promoter$ and 524,816 for $TF\_enhancer$), two between regulatory regions and transcripts weighed by the $distance$ and the $Weight\_Distance$ between those entities (123,441 entities for $promoter\_transcipt$ and 950,514 entities for $enhancer\_transcipt$) and a last one between transcripts and genes (53,449 entities). Each instance of classes $Promoter$ or $Enhancer$ is associated with two sets of 808 float values, one corresponding to its expression value in every sample, and the other corresponding to its normalized relative rank in each sample compared to the 807 others. Similarly, each instance of the $Transcript$ class is associated with 808 float values, describing its normalized relative rank in each sample compared to the others. This rank information is directly provided by *Regulatory Circuits*. Contrary to the promoters and enhancers, no measured expression value is provided for transcripts. Each rank identifier is built by using the sample's identifier (*libId*).

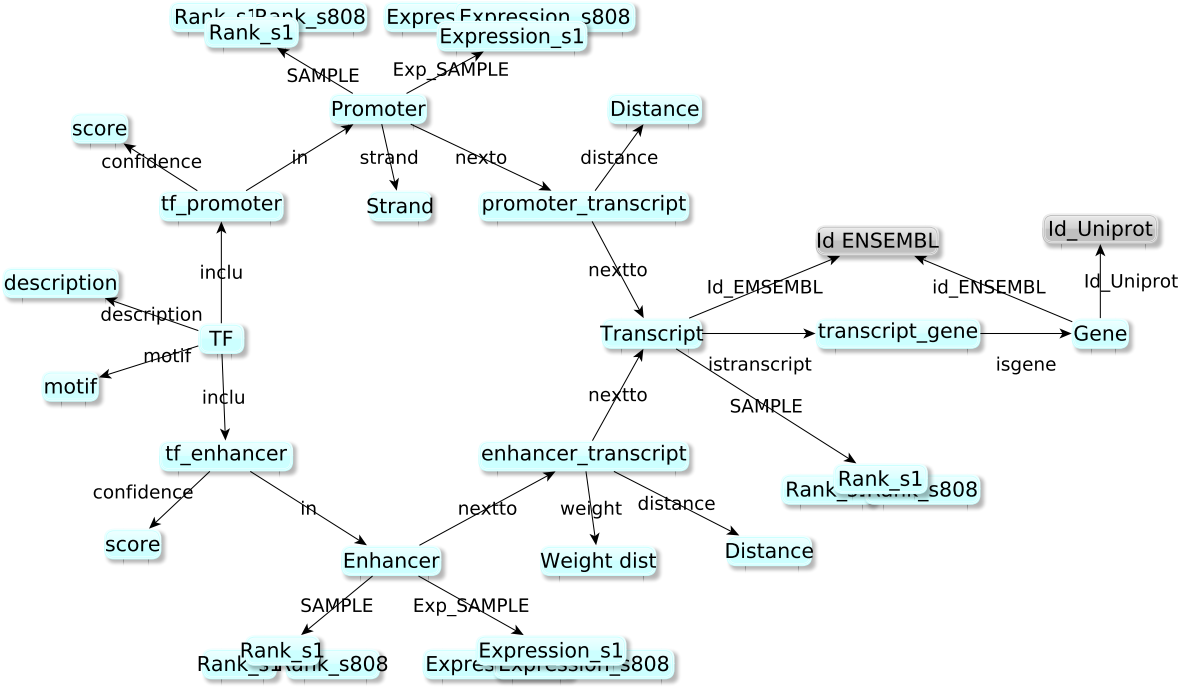


FIGURE 5.3 – Structure of the graph of biological data from the *Regulatory Circuits* project. Boxes represent classes of entities. The grey boxes represent mappings to external resources.

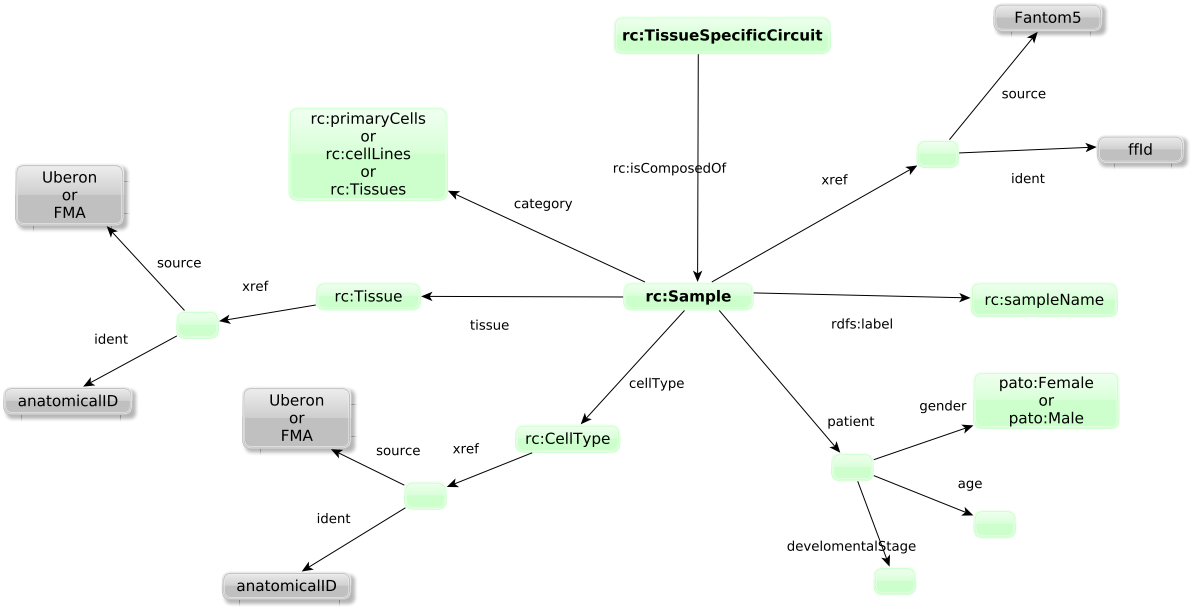


FIGURE 5.4 – Structure of the graph of experimental conditions describing the samples and the tissue-specific circuits (in bold). Boxes represent classes of entities. The grey boxes represent mappings to external resources.

The graph of experimental conditions (FIGURE 5.4) describes the experimental information about the 808 samples (cell types, organs, patient, diseases... and mappings to reference databases such as Uberon) and the 394 tissues.

The graph of metadata contains the VoID descriptions of each of the other graphs. For the sample-specific and tissue-specific regulatory networks, these descriptions refer explicitly to the sample or tissue from the graph of experimental conditions.

5.3.3 Sample-specific weights of the TF-gene regulations

According to *Regulatory Circuits* published methodology, the TF-gene interactions are mediated by the ability of the TF to bind into regulatory regions of the chromatin (enhancers or promoters), the distance of this region to the gene (enhancer being farther and promoter being adjacent to the genes), the region accessibility and the gene expression. Each TF-gene interaction is therefore characterized by a promoter weight and by an enhancer weight. As shown in FIGURE 5.3, the relation between the

TF and a regulatory region is described by a $confidence$ value, and the $rank$ of the regulatory region is described by a value associated with the sample. The promoter weight is defined by $weightP = \max((confidence \times rank\_promoter\_sample)^2)$, where the maximum is computed for all the possible promoters mediating the interaction. The enhancer weight is defined by: $weightE = \max((confidence \times Weight\_Distance \times \sqrt{(Rank\_transcript\_sample \times Rank\_enhancer\_sample)})^2)$.

For each of the 808 samples, a distinct named graph represents these sample-specific weights characterizing the TF-gene regulation relations by reified relations (FIGURE 5.6 left). The SPARQL queries introduced in [LOUARN et al., 2019] are adapted to compute the weights, and an INSERT operation is added to re-inject the result into the sample-specific graphs. The relations with a null weight are excluded to avoid overloading the graph. The SPARQL query for computing $weightP$ is given in FIGURE 5.5, where *SAMPLE* must be replaced by the identifier of an actual sample. A similar query for computing $weightE$ is available on the Github repository of the project [1]. In total, the sample-specific graphs contain 888,602,040 triples.

5.3.4 Tissue-specific weights and score of the TF-gene regulations

According to the experimental data graph, each tissue is associated to 1 to 33 samples.

The tissue-specific TF-gene interaction network is obtained by aggregating the sample-specific networks. As in sample-specific networks, TF-gene relations are characterized by (i) a promoter weight (class $WeightP$), (ii) an enhancer weight (class $WeightE$), and (iii) a global score (which is the max of the two weights, class $Score$) that corresponds to the only TF-gene relation score given in the published regulatory networks of *Regulatory Circuits*. In a tissue, the $WeightP$ and $WeightE$ values of a TF-gene relation are obtained as the maxima of the corresponding weights of the same relation in the RDF graphs specific for all the samples constituting the tissue.

For each of the 394 tissues, a distinct named graph represents these tissue-specific weights and scores characterizing the TF-gene regulation relations by reified relations (FIGURE 5.6 right). A SPARQL query is designed to compute tissue-specific

1. https://github.com/mlouarn/RCsparql

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX user: <http://regulatorycircuits.org/data/>
PREFIX graph: <http://regulatorycircuits.org/graph/>
PREFIX rco: <http://regulatorycircuits.org/ontology/>

INSERT {
  GRAPH graph:SAMPLE {
    _:idRel rco:fromTF ?tf_uri .
    _:idRel rco:fromGene ?gene_uri .
    _:idRel rco:weightP ?max_weightP .
  }
}

WHERE {
    SELECT ?tf_uri ?gene_uri (max (?weightP) AS ?max_weightP)
        WHERE {
            ?tf_uri rco:inclu ?tf_promoter_uri .
            ?tf_promoter_uri rco:in ?promoter_uri .
            ?promoter_transcript_uri rco:nextto ?promoter_uri .
            ?transcript_uri rco:nextto ?promoter_transcript_uri .
            ?transcript_gene_uri rco:istranscript ?transcript_uri .
            ?gene_uri rco:isgene ?transcript_gene_uri .
            ?tf_uri rdf:type user:tf .
            ?tf_uri rdfs:label ?tf_Label .
            ?tf_promoter_uri rdf:type user:tf_promoter .
            ?tf_promoter_uri rdfs:label ?tf_promoter_Label .
            ?tf_promoter_uri user:confidence ?tf_promoter_confidence .
            ?promoter_uri rdf:type user:promoter .
            ?promoter_uri rdfs:label ?promoter_Label .
            ?promoter_uri user:SAMPLE ?promoter_SAMPLE .
            ?promoter_transcript_uri rdf:type user:promoter_transcript .
            ?promoter_transcript_uri rdfs:label ?promoter_transcript_Label .
            ?transcript_uri rdf:type user:transcript .
            ?transcript_uri rdfs:label ?transcript_Label .
            ?transcript_uri user:SAMPLE ?transcript_SAMPLE .
            ?transcript_gene_uri rdf:type user:transcript_gene .
            ?transcript_gene_uri rdfs:label ?transcript_gene_Label .
            ?gene_uri rdf:type user:gene .
            ?gene_uri rdfs:label ?gene_Label .
            BIND (xsd:float(?tf_promoter_confidence) * xsd:float(?tf_promoter_confidence) *
            xsd:float(?promoter_SAMPLE) * xsd:float(?promoter_SAMPLE) AS ?weightP).
            FILTER ( ?weightP > 0 )
        }
}
```

FIGURE 5.5 – SPARQL query for computing the sample-specific value of the weight associated to promoters for the TF-gene regulation relations and inserting it in the corresponding sample graph

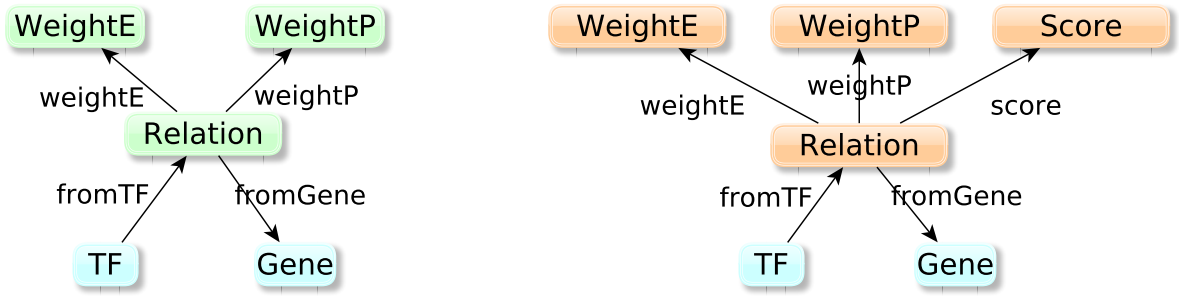


FIGURE 5.6 – Structure of the 808 sample-specific (left) and 394 tissue-specific (right) graphs representing the parameters of the TF-gene regulation relations. Boxes represent classes of entities.

weights and scores and re-inject them into the tissue-specific RDF graphs. The query in FIGURE 5.7 computes weights of TF-Gene relations in a tissue-specific network formed by two separate Samples. Queries for tissue-specific network with more samples or a single sample are available in the Github repository of the project [2]. In total, the tissue-specific graphs contain 916,758,018 triples.

5.3.5 Overall dataset

All data related to *LERC* are available on the website of the project [3].

The overall dataset is composed of 1205 RDF named graphs. The resource is composed of 2,145,789,028 triples and the distribution of the triples by graphs can be seen in TABLE 5.1.

In total it required 28.6 days CPU to generate the dataset from the initial integration of the biological data graph to the computation of the tissue-specific graphs. TABLE 5.1 compiles the total number of triples, entities and classes in the biological data graph. The population of the sample and tissue-specific graphs is described in TABLE 5.1. On average a sample-specific graph is composed of 1,099,755 triples and a tissue-specific graph is composed of 2,326,797 triples.

2. https://github.com/mlouarn/RCsparql
3. https://regulatorycircuits-lod.genouest.org

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX user: <http://regulatorycircuits.org/data/>
PREFIX graph: <http://regulatorycircuits.org/graph/>
PREFIX rco: <http://regulatorycircuits.org/ontology/>
INSERT { GRAPH graph:TISSUE {
    _:idRel rco:fromTF ?tf .
    _:idRel rco:fromGene ?gene .
    _:idRel rco:weightP ?wp .
    _:idRel rco:weightE ?we .
    _:idRel rco:score ?score .
  } }
WHERE {
  VALUES ?g1 {graph:SAMPLE1}
  VALUES ?g2 {graph:SAMPLE2}
  ?tf rdf:type user:tf .
  ?gene rdf:type user:gene .
  OPTIONAL {
    GRAPH ?g1 {
      ?rP rco:fromTF ?tf.
      ?rP rco:fromGene ?gene .
      ?rP rco:weightP ?weightP1 .
    }
  }
BIND (IF(bound(?weightP1), ?weightP1, "0.0"^^xsd:float) AS ?Wp1) .
  OPTIONAL {
    GRAPH ?g1 {
      ?rE rco:fromTF ?tf.
      ?rE rco:fromGene ?gene .
      ?rE rco:weightE ?weightE1 .
    }
  }
BIND (IF(bound(?weightE1), ?weightE1, "0.0"^^xsd:float) AS ?We1) .
  OPTIONAL {
    GRAPH ?g2 {
      ?rP rco:fromTF ?tf.
      ?rP rco:fromGene ?gene .
      ?rP rco:weightP ?weightP2 .
    }
  }
BIND (IF(bound(?weightP2), ?weightP2, "0.0"^^xsd:float) AS ?Wp2) .
  OPTIONAL {
    GRAPH ?g2 {
      ?rE rco:fromTF ?tf.
      ?rE rco:fromGene ?gene .
      ?rE rco:weightE ?weightE2 .
    }
  }
BIND (IF(bound(?weightE2), ?weightE2, "0.0"^^xsd:float) AS ?We2) .
BIND (IF((?We1 > ?We2),?We1, ?We2) AS ?We)
BIND (IF((?Wp1 > ?Wp2),?Wp1, ?Wp2) AS ?Wp)
BIND (IF((?We > ?Wp),?We, ?Wp) AS ?score)
FILTER (?score >0)
}
```

FIGURE 5.7 – SPARQL query for computing a tissue-specific values of the weights associated to promoters and enhancers and the global score for the TF-gene regulation relations from the values of the samples associated to the tissue, and inserting them in the corresponding tissue graph

TABLE 5.1 – Characteristics of the RDF dataset

	Number of elements
Triples	340,428,970
Entities	3,226,341

(a) Integrated data in the biological data graph before running the injection queries

Nb of triples	By samples Promoter	By samples Enhancer	By tissues
Minimum	408,618	146,196	909,670
Maximum	592,698	1,282,083	5,106,105
Average	513,644	586,111	2,326,797
EST. TOTAL	415,024,352	473,577,688	916,758,018

(b) Population of the re-injected graphs

Time	By samples Promoter	By samples Enhancer	By tissues
Minimum	4m26.779s	7m35.194s	24m29.327s
Maximum	7m5.927s	21m25.270s	120m04.794s
Average	5m30.071s	15m29.902s	59m53.039s

(c) Execution times of the queries, calculated for the first 102 samples and 55 tissues networks

5.3.6 Biologically-relevant queries

As we have seen, exposing the intermediate results such as the sample-specific regulation networks allows biologists to access the information they need.

Moreover, all this additional information allows biologists to tailor their analysis according to their needs. For example, the *Regulatory Circuits* dataset has a tissue-specific circuit “CD14+ Monocytes" composed of 33 samples. However, among these samples, only 3 were measured in blood (for the others, the tissue of the measure is unknown), in men aged 47, 57 and 53. If the biologist is specifically interested in CD14+ Monocytes from blood for man over 55, the graph of experimental data can be queried to create a dedicated new tissue with the following query (FIGURE 5.8). From there, the metadata graph is queried to retrieve the identifiers of the sample-specific graphs, and then the query presented in FIGURE 5.7 is run to compute the weights and score of the regulation relations in this new tissue, thus extending *Regulatory Circuits* by reusing a part of its data.

More examples of queries are available on the Github of the project [4].

4. https://github.com/mlouarn/RCsparql/tree/master/queries_examples

```
FROM graph:experimentalContext

INSERT {
  GRAPH graph:experimentalContext {
    rc:myTissue rdf:type rco:TissueSpecificCircuit .
    rc:myTissue rco:isComposedOf ?sample .
  }
}
WHERE {
  rc:CD14%2B%20Monocytes rco:isComposedOf ?sample .
  ?sample rc:tissue rc:blood .
  ?sample rc:patient [ rc:gender pato:Male ;
                       rc:age ?age ] .
  FILTER ( ?age > "55"^^xsd:integer)

}
```

FIGURE 5.8 – SPARQL query for retrieving the set of samples that meet some conditions expressed by the user (here, identify the subset of the “CD14+ Monocytes" samples taken in the blood of male patients over 55 years old) and creating the associated user-defined tissue

5.4 Discussion and perspectives

In this chapter, we address the issue of allowing the reusability of the existing source biological datasets from the *Regulatory Circuits* project [MARBACH et al., 2016] and of enriching its published output, which consists in 394 TF-gene interaction (large-scale) tissue-specific networks. These networks result from the aggregation of 808 sample-specific TF-gene interaction networks, which were unpublished and are also integrated in the dataset we propose.

The RDF representation of the *Regulatory Circuits* dataset follows the best practices, using reification entities for weighted relations, using named graphs, and follows the FAIR guidelines. It also reuses already suitable existing resources class such as Uniprot and Ensembl identifiers and follows the faldo chromosomal localization format. Federated SPARQL queries can then be used to combine information for *Regulatory Circuits* with information from these resources (e.g. associations with diseases, or annotations).

LERC is available on a persistent domain and every queries are publicly available on Github. The original datasets can be downloaded as tabulated files from the website of the original project [5].

5. http://regulatorycircuits.org/download.html

By converting the *Regulatory Circuits* dataset into RDF with our modular principles, our contribution to reusability is threefold. First, it *facilitates the reuse of Regulatory Circuits's results in other studies* by providing access to the tissues-specific regulatory networks and the associated information. Second, it *facilitates the reuse of the studies' data in other pipelines* by providing access to the samples' experimental context and to the intermediary results such as the sample-specific regulatory networks, which can be reused to compute other indicators than *Regulatory Circuits* weights and scores. Third, it *provides the capacity to enrich the Regulatory Circuits dataset with additional information* as the data model and Semantic Web technologies support adding new samples or defining new tissues, and the SPARQL queries we provide generate the corresponding weights and scores. Overall, the *Regulatory Circuits* case study confirms that Semantic Web technologies are a relevant solution for reusing knowledge bases [KAMDAR et al., 2019 ; S. STEPHENS et al., 2006], and demonstrates that they are also applicable to address the challenge of integrating them to project-specific datasets [H. CHEN et VANBUREN, 2012].

Improving the exploration of *Regulatory Circuits* biological data and networks In a previous work we showed that *Regulatory Circuits* workflow could be described using Semantic Web technologies thus increasing its reproducibility. This new implementation, including not only the input data but also the TF-gene interaction networks resulting from the in-silico integration of source biological data, improves the browsing of this resource. The implementation we propose allows a fine-grained exploration: the user can select part of the network, for example excluding regions at a lower distance than the *Regulatory Circuits* threshold, or excluding one type of region (e.g. for taking into account that promoters relations are more reliable).

Improving the reusability and the enrichment of the source biological data with SPARQL queries The networks available in the *Regulatory Circuits* website are static and do not evolve with the biological datasets it was based upon. A major advantage of our approach is that TF-gene interaction networks for samples and tissues are generated with SPARQL queries from the source biological data before being inserted in the resource. This implementation allows a user to easily change some bricks of the integration pipeline which generates the network, such as new calculation of the ranks, adding new genes or transcription factors in the networks, or removing some of

them: all these changes will be translated into modifications of the few queries used to generate the network.

Our resource and the approach we use to populate it also facilitates the generation of new TF-gene interaction networks for new tissues, through the aggregation of samples with different characteristics than those chosen in the *Regulatory Circuits* project. Indeed, *Regulatory Circuits* present 394 tissue-specific networks, but looking into the detail of the sample and tissues correspondence evidenced some tissues that could be separated into smaller sets. For example, the “CD14+ Monocytes" network given in *Regulatory Circuits* is based on 33 CD14+ monocytes cell samples, which have different characteristics (origin, donor age...). The modular structure of our resource allows for the computation of new TF-gene interaction network using these characteristics to discriminate the samples.

Finally, using our RDF resource of *Regulatory Circuits* introduced in this chapter and the strategy used to build it allows a user to add new tissues or TFs if they have similar input data. This would require to pre-compute rankings for transcripts and regulatory regions which are at the moment provided by the *Regulatory Circuits* resource and cannot be recomputed. Similarly, introducing a new TF would require to introduce new confidence values for its binding in regulatory regions.

Improving interoperability Among the 150 articles citing *Regulatory Circuits*, at least 42 either use directly the resulting networks for biological data explanation or use them as comparison for regulatory network inference. And, in 10 of these, *Regulatory Circuits* was used in combination with one or several other databases. Other resources on TF-genes relations exist [HAN et al., 2018][LICATA et al., 2020] but are complementary of *Regulatory Circuits*, the latter being the only one categorizing tissue-specific networks. By representing *Regulatory Circuits* as an RDF graph we therefore improve its interoperability with resources of similar scope already using Semantic Web technologies and helps its reuse in combination with other already existing RDF resources. In particular, it significantly extends the part of FANTOM5 data available as RDF [ABUGESSAISA et al., 2016 ; LIZIO et al., 2015].

A generic approach for the enrichment of source biological data with the result of data-analyses Our results show that a Semantic Web approach scales not only for

the integration of large-scale biological data but also for the iterative enrichment of such a resource with the results of in-silico analyses modeled with SPARQL queries. This is possible by using a modular structure based on named graphs. This strategy could be easily transposed to other life science studies which analysis pipeline describes relations with simple arithmetic functions. For more in depth analyses, this approach could be transposed while pre-processing the most complex computing tasks. Our work supports the adoption of Semantic Web technologies as it is a large real data graph, which is used in several studies and does have an impact in life science before integration.

5.5 Conclusion

As we shown in this chapter and in the previous one, Semantic Web technologies are a relevant tool for regulatory networks data-bases. The lower layers of the Regulatory Circuits workflow: obtaining sample-specific networks can be reduce to two well performing queries. But we add in complexity when trying to re-inject the result of those queries to further extract the networks in higher level such as tissue-specific networks. As we seen the re-injection of the 808 samples and 394 tissue-specific networks took almost a month of computation.

But the end result is a triple-store wildly available, containing all the resources necessary to either reuse the result of Regulatory Circuits - even the intermediary ones not given in the original downloadable folder - or to add new information to reproduce the workflow with additional data.

CHAPITRE 6

DESIGN OF A SUITABLE PIPELINE FOR BIOLOGICALLY-CLOSE AND SPARSE CELLS TYPES

This chapter goal is to present a new design of pipeline to infer new regulatory networks, with a focus on small sets of closely-related cells types.

In SECTION 6.1 we present the overall design of this pipeline, what does it take as entry and what information we capitalised on from other resources. SECTION 6.2 focus on the pre-processing of the data-sets use for the pipeline: creation of patterns to regroups genes and regions of similar expression across the cells populations, finding the TF binding sites in the given regions and calculation the closeness for which the regions can regulate the genes. Then, in SECTION 6.3 we explain the data-graph created from those files and how we integrated it to later query the data-graph and extract regulatory networks. SECTION 6.4 present the post-processing where we look at the relations infer in the previous section and checks their consistency with the biology to asses if they can or not have a regulatory impact on the network. This step act as a filter. SECTION 6.5 present the automation of the pipeline, compiling all the previous steps in a snake-make and python implementation. Finally, in SECTION 6.6 we compare the design of our pipeline to the workflow of *Regulatory Circuits* - presented in CHAPTER 3 and in the following SECTION 6.7 we run our pipeline on data-set extracted from *Regulatory Circuits* and look at the resulting networks in comparison of *Regulatory Circuits*.

6.1 Introduction

As presented in CHAPTER 3, we raised several issues on existing network inference workflows, and especially on *Regulatory Circuits* itself. As synthesized in FIGURE 6.1, the issues span the files formats, the workflow description, inconsistencies between the workflow description and the provided intermediary files, as well as methodological problems. In this chapter we introduce a new pipeline addressing the issues on the methodology (in green in the figure). This pipeline (1) takes into account the activities of the TFs themselves, (2) proposes an alternative to the discretization of the activities as ranks and (3) produces signed networks.

We propose a regulatory-network inference pipeline. It has been developed to be stringent and to limit the space of the candidates TF-genes relations. The candidate relations are the most likely to occur but will need to be confirmed through biological experiments or bibliography review.

6.2 Design

The goal of the pipeline is to find the regulators of genes sets of similar expression and to qualify these TF-genes' relations as either activation or inhibition.

This pipeline requires: a list of genes, their expression in a selected number of cell types, their genomic coordinates and a list of selected regulatory regions with their activity and genomic localisation. For genes and regions the coordinates are determined according to the GRCh37 or hg19 [MEYER et al., 2013] [KAROLCHIK et al., 2014] human reference genome from UCSC [KENT et al., 2002].

The pipeline also needs TF binding sites localisation. This can be provided by the user, but we provide an implementation using the genome-wide TF binding sites coordinates from FANTOM5 [ANDERSSON et al., 2014] [MARBACH et al., 2016] (Functional Annotation of the Mammalian Genome).

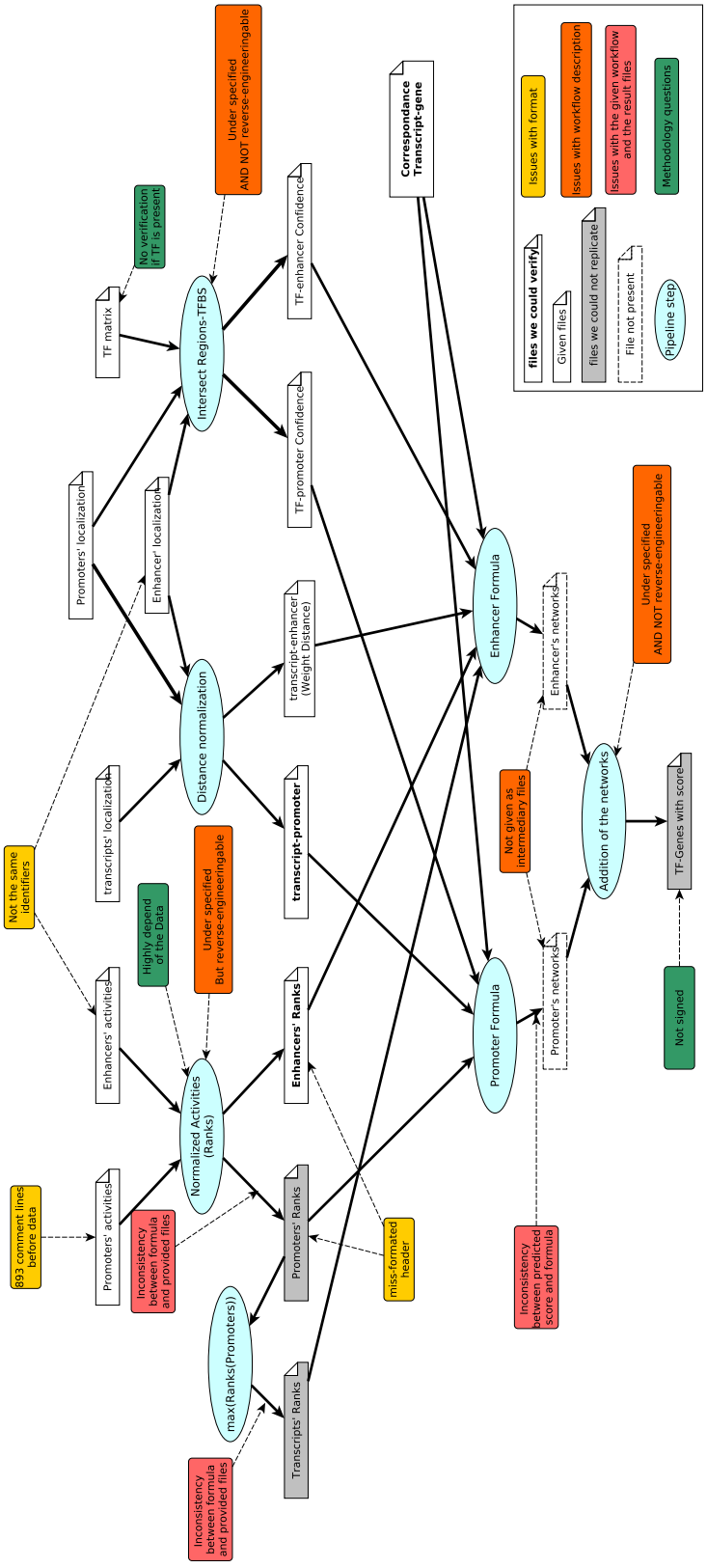


FIGURE 6.1 – *Regulatory Circuits* global workflow with the addition of the point of contention: either issues in the workflow or remaining doubt on the step.

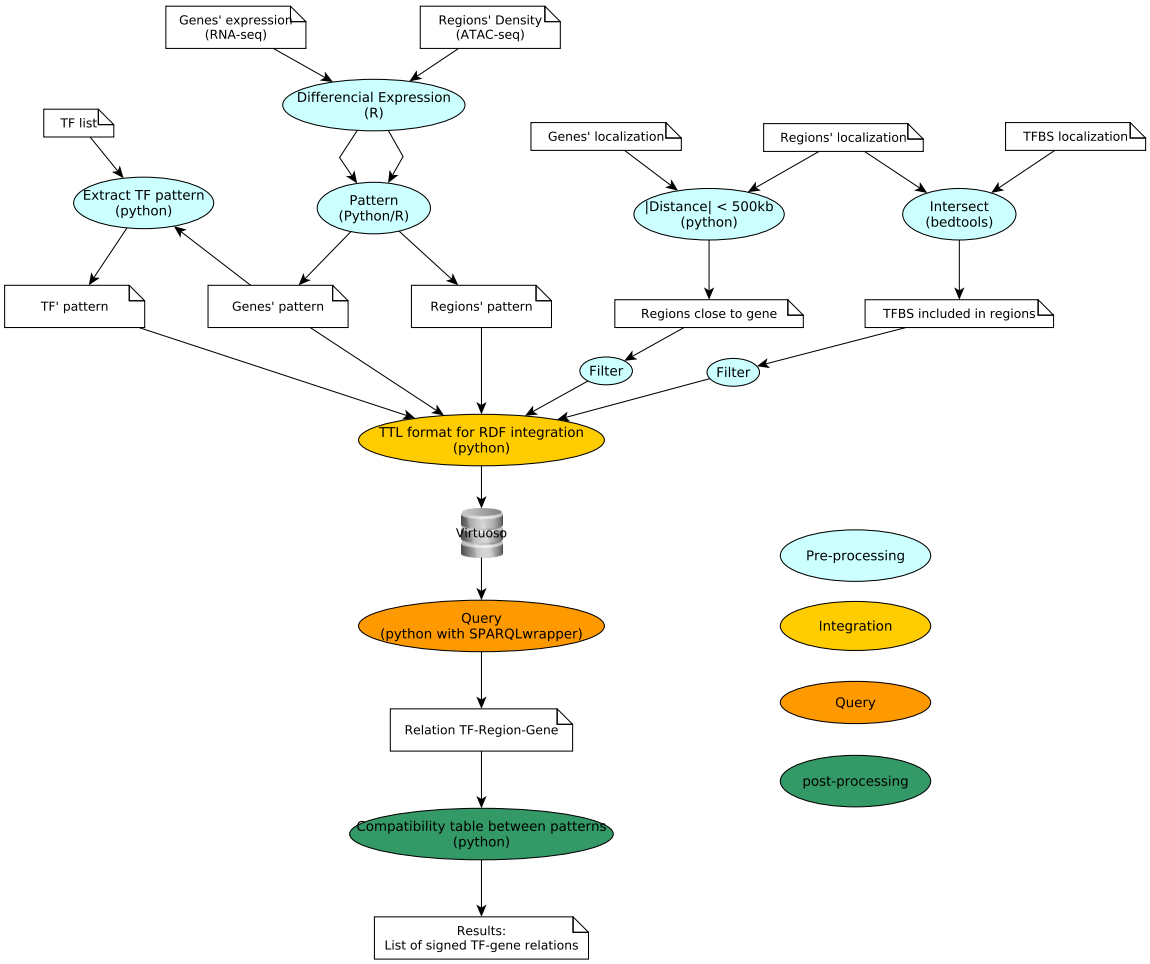


FIGURE 6.2 – Representation of the different steps of this pipeline.

FIGURE 6.2 presents the different steps of the pipeline: the transformation of individual expression in patterns, finding the relations between genes and regions and finding the TF binding sites in the provided regions. The pre-processed data are then integrated using Semantic Web technologies. They can then be queried to find the candidate relations between TF, regions and genes that respect a given set of rules: the regions must be at most at 500 kb of the gene, the TF must have a binding site in the region. Query results must then be refined to identify the TF-gene relations that are compatible with their expressions and assigned them signs (either positive, indicating an induction, or negative, indicating an inhibition).

As an output, the pipeline gives a list of candidates signed TF-genes relations that can be explored to find new regulators.

6.3 Pre-processing

FIGURE 6.3 represents the different steps of data pre-processing. The top of the figure represents the background knowledge on TF binding sites, bellow is represented the input data: regulatory regions (their coordinates and read density), genes (coordinates and expression level). The bottom represents the processes to actually integrate these data. The first step discretizes of gene expressions and regulatory regions read densities into patterns. The second step computes the distances between neighbor genes and regions. The third step finds the inclusion of TF binding sites into regions.

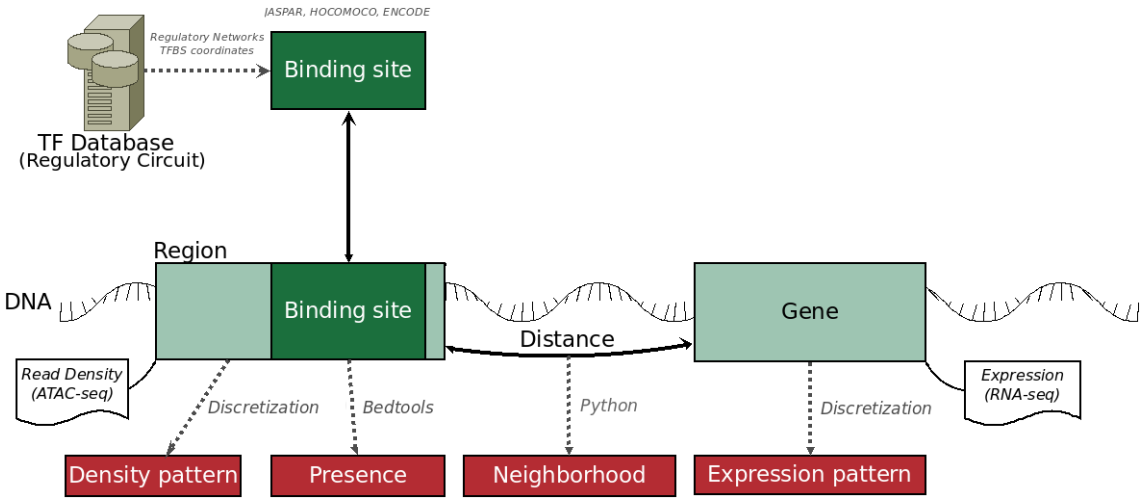


FIGURE 6.3 – Representation of the different pre-processing steps (in red) of our pipeline.

6.3.1 Discretization patterns for read densities and gene expression

The idea behind the pattern is to be able to regroup genes of similar behavior in the different populations. A solution could have been to use co-expression analysis for example using the WGCNA R [LANGFELDER et HORVATH, 2008] package. Unfortunately, after testing, it gave poor results with limited data-sets such as ours.

The patterns do not show a chronological order between the different populations, which can be placed in any order.

Preliminary analysis

The first step of discretization of the expression and density, is to identify features which are relevant for building a regulatory network, i.e. those which vary between the compared cell types. We ask of the user to provide a list of differently expressed genes and regions, on which patterns will be generated.

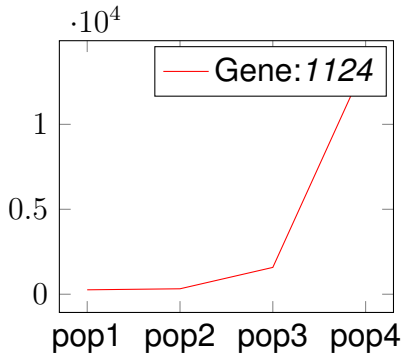
The user can also provide a list of genes with low expression in all populations and a list of genes with similar expression in all population that will both be affected with specific patterns.

Gene Expression Pattern

The discretization has been performed independently for each gene. A gene expression pattern or gene expression profile is based on a several digits pattern, one for each cell population (ex: *pop1: 1, pop2: 1, pop3: 1, pop4: 4 is pattern: 1114* or another pattern is *1234*) each digit having a value ranging from 1 to 4, leading to a potential of 256 distinct patterns. To determine this pattern we first compute the mean per populations based on normalized count from the differential expression analysis. We used a logarithmic discretization on these average values. Furthermore, the population with the lowest expression gets always *1* in the profile and the population with the highest always gets *4*. We generated the expression profiles with the R software and using a custom function that affects its class number to each element of a distribution when the number of classes is specified. On four populations, once these pattern were discretized, we obtained 126 potential different profiles. This is explained by our principles which favor extreme patterns: *1114* will be kept instead of *1113* and *4441* instead of *4442* and impose at least a *1* and a *4* in each pattern.

Finally, genes with low expression as defined in differential expression analysis have been granted the profile *0000* and have been removed from the implementation. Genes with constant expression in all population have been granted the profile *5555*.

The TF patterns are those of their respective coding genes.



(a) Graphical representation of a gene expression

	pop1	pop2	pop3	pop4
Biological data	259,61	319,72	1578,61	13589,51
Pattern	1	1	2	4

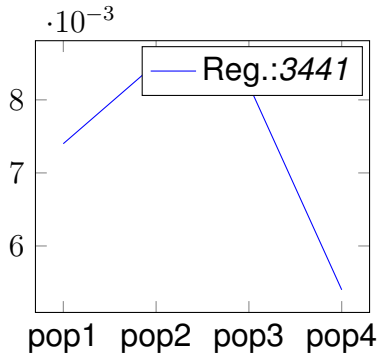
(b) Expression of a gene and expression pattern *by cell type*

FIGURE 6.4 – Example of a gene expression and pattern: This gene expression is characterized by three low expression values followed by an higher one. This expression is modeled by the 1124 pattern, favoring extreme values.

Region Density Pattern

Similarly to the gene expression pattern, the region density pattern is a pattern composed of digits corresponding to the number of populations, each point from 1 to 4, based on the read densities of each region. This profile is also determined using a logarithmic discretization and implemented in R.

The read density patterns have been defined by discretization for a union of unique regions from all experiments. This discretization led to a pattern following the same properties than the ones for the expression pattern. An example with a region can be seen in FIGURE 6.5, where we can see that this region has a low read density in the population pop4 but a high density in the populations pop2 and pop3. For the regions, the profile *0000* does not appear, as it is necessary for the region to be detected in at least one population to be selected.



(a) Graphical representation of a region density pattern

	pop1	pop2	pop3	pop4
Read density	0.0074009	0.00859515	0.00822473	0.00543647
Pattern	3	4	4	1

(b) Read density of a region and density pattern *by cell type*

FIGURE 6.5 – Region read density and pattern: the expression of this region is characterized by one medium values, two high read density values followed by a very low one. This expression is modeled by the 3441 pattern, favoring extreme values.

6.3.2 Neighborhood relationship

The second step was to define the neighborhood relationship between a region and a gene. If a region is relatively close to a gene on a same chromosome, it can be a regulatory region of this gene if presenting TF binding site motifs.

The implementation was done in python. It uses as entry the genomic coordinates of both regulatory regions and genes. We computed the relations between regions and genes keeping all relations which distance was inferior or equal to 500 kb (using the same limit as Regulatory Circuits). The region was kept regardless of its position to the gene (before or after). The distance was calculated between the two closest extremities of the entities. If the region and the gene are overlapping, either included or exceeding, the distance is set to 0.

As some genes have duplicates, we only kept the smallest distance between a gene and a region, regardless of the duplicate use.

6.3.3 Finding TF binding sites in our regions

We decided to use the *Regulatory Circuits* [MARBACH et al., 2016] data on TF localization across the genome. We used the genomic coordinates of the TF binding

sites identified in the Regulatory circuits data [1] which is based on FANTOM5. The integration of the TF localization was at first processed using *Bedtools intersect* [QUINLAN, 2014] tool, looking for all the binding sites included into our regions. We kept the presence information only: if a TF had a binding site in a region we kept the information, but if it had more than one binding site only kept one, not all the binding site localization nor their number. This was done to reduce the size of the output file and to optimize the time of the latter queries.

6.4 Data graph for the integration

We integrated the discretized patterns into an RDF graph, to do so a first step was to ensure that all the necessary files were formatted for the integration. Each entity needs to present a unique identifier, identical across all files to allow to refer specifically to this entity. For the regions, we used an identifier designed after the type of region (i.e. ATAC_; and Region_ for the rest of this chapter) followed by the row number at which they appear in the region localisation file - this ensured that a same number was never used twice for different entities. For genes and TF, we kept the usual names (HGNC Gene Symbols) as identifiers.

To implement the weighed relations (distance and tf_inclusion), we needed to add reified entities as RDF does nor allow relations to bear a score. We defined two reified entities: Region_closest (weight = integer) and TF_inclusion which correspond to the last two steps of the pre-processing.

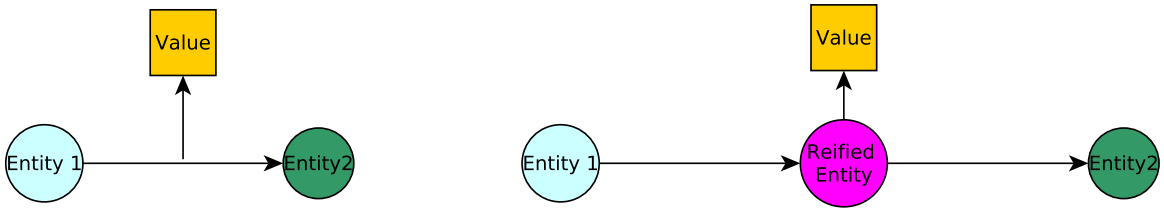


FIGURE 6.6 – Example of reification between the entities Entity1 and Entity2, creation of the Reified Entity whom bear the value of the relation.

The data model structure after integration is illustrated in FIGURE 6.7. This figure

1. http://regulatorycircuits.org/download.html

can be seen as a representation of the interactions between the data, where the entities are linked between each other by different relationships.

To retrieve TF-Gene relations and the necessary patterns all the entities presented in FIGURE 6.7 are not necessary, some of them have been added to help refine the results. The entities that are strictly necessary are: genes, TF and regions with their respective patterns, as well as the reified entities Region_closest and TF_inclusion. We choose to add references to Uniprot and Ensembl for the genes as it can add information to the graph. We also choose to add the localisation information that we used in previous steps, as a potential filtering parameter.

From this question and the data structure we used AskOmics to generate the query shown in FIGURE 6.8: starting from the green node "Gene" having an expression pattern, we trace back all ID_ATAC (red node) which are connected to the Gene by a ATAC_closest relationship (orange node). We filter these Id_ATAC by taking into account the transcription factor (cyan node) which have a relationship with the Id_ATAC through the Id_Inclusion (dark blue node), representing the presence of a binding site. Along the way, we gather the patterns for the TF, the region and the gene, as we will need them in the next step. This automatically generate the SPARQL query presented in FIGURE 6.9.

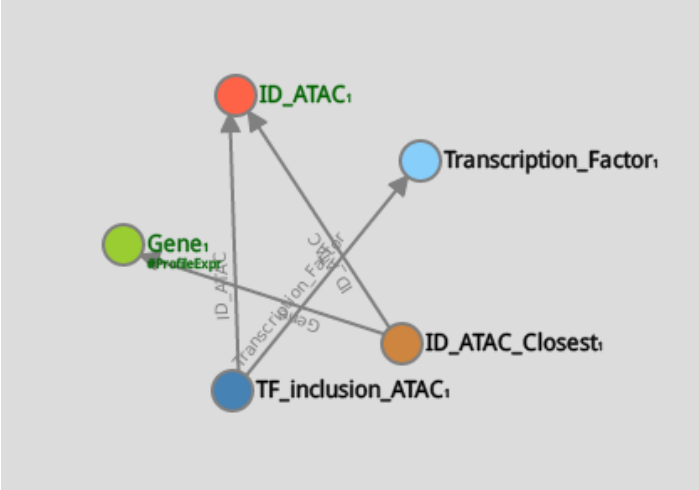


FIGURE 6.8 – The final AskOmics query: *starting from a gene of a given profile then look for its neighbor regions and then for the TF included in these regions.*

This query was challenging to run directly in AskOmics - as the length of the result was automatically cut after an arbitrary number and sometime produced time-out while

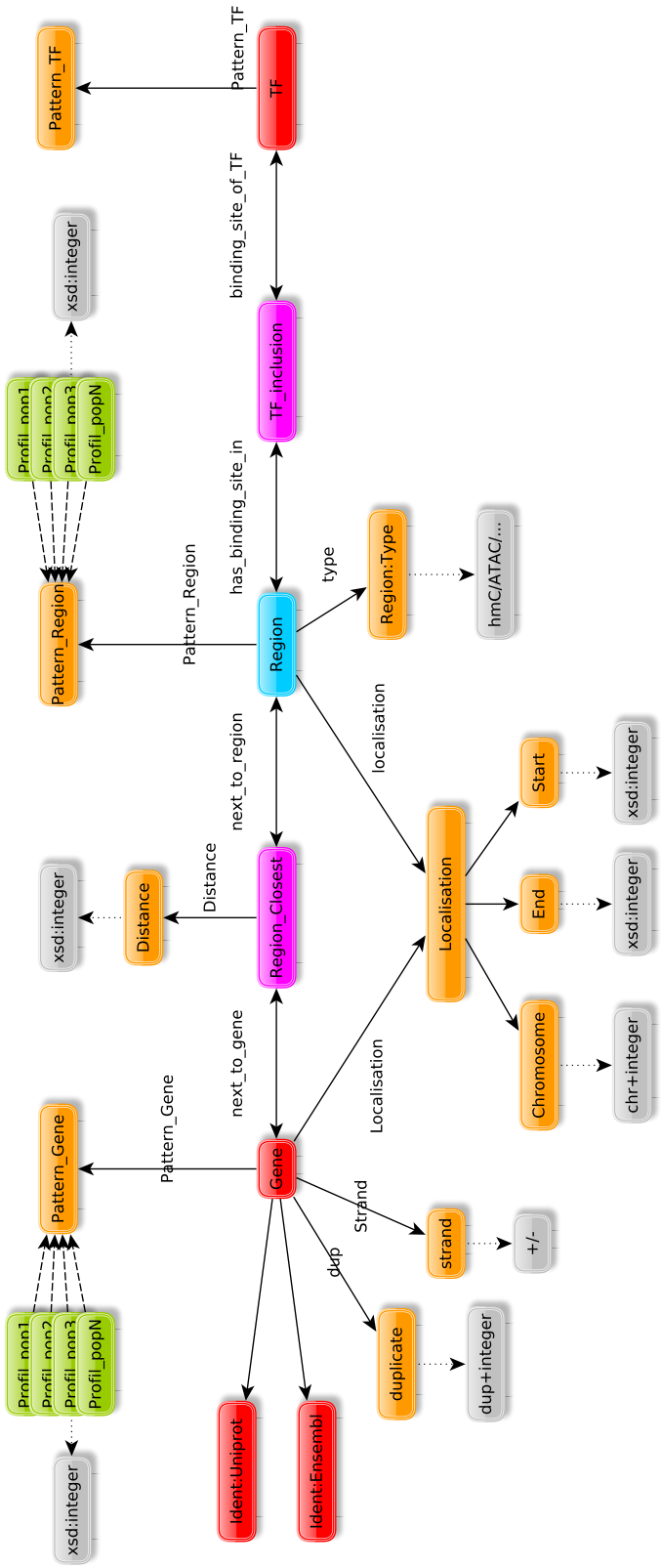


FIGURE 6.7 – Data model: *Representation of the data integrated in AskOmics and their dependencies.*

```
PREFIX : <http://www.semanticweb.org/user/ontologies/2018/1#>
PREFIX askomics: <http://www.semanticweb.org/askomics/ontologies/2018/1#>
PREFIX faldo: <http://biohackathon.org/resource/faldo/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT DISTINCT ?Gene ?Pattern_Gene ?Region ?Pattern_Region ?TF ?Pattern_TF
WHERE {
    ?Gene_uri :next_to_gene ?Region_Closest_uri .
    ?Region_Closest_uri :next_to_region ?Region_uri .
    ?TF_inclusion_uri :has_binding_site_in ?Region_uri .
    ?TF_inclusion_uri :binding_site_of_TF ?TF_uri .
    ?Gene_uri rdf:type :Gene .
    ?Gene_uri rdfs:label ?Gene .
    ?Gene_uri :PatternGene ?Pattern_GeneCategory .
    ?Pattern_GeneCategory rdfs:label ?Pattern_Gene .
    ?Region_Closest_uri rdf:type :Region_Closest .
    ?Region_Closest_uri rdfs:label ?Region_Closest .
    ?Region_Closest_uri :Distance ?Region_Closest_Distance .
    ?Region_uri rdf:type :Region .
    ?Region_uri rdfs:label ?Region .
    ?Region_uri :Pattern_Region ?Pattern_RegionCategory .
    ?Pattern_RegionCategory rdfs:label ?Pattern_Region .
    ?TF_inclusion_uri rdf:type :TF_inclusion_ATAC .
    ?TF_inclusion_uri rdfs:label ?TF_inclusion .
    ?TF_uri rdf:type :Transcription_Factor .
    ?TF_uri rdfs:label ?TF .
    ?TF_uri :PatternTF ?Pattern_TFCategory .
    ?Pattern_TFCategory rdfs:label ?Pattern_TF .
    FILTER ( ?Region_Closest_Distance < 500000 ) .

}
```

FIGURE 6.9 – SPARQL query for retrieving all relations between TF-Region-Gene and their associated patterns. This code was automatically generated by AskOmics from the graphical query shown at FIGURE 6.8

reaching to the platform hosting it - and required to wrap the SPARQL query (FIGURE 6.9) into Python. An example of the results of this query is shown in TABLE 6.1, forming a triple between the gene, the neighbor region and the TF with a binding site in said region, and their mutual expression and density patterns. But all the TF found in the triples may not have a regulatory impact: it depends on their expression and the accessibility of the chromatin at regions containing their binding sites (i.e. TF pattern equal to 0, region and TF activities contradictory,...). This is the reason we developed a tool to checks those rules, as explained in the next section.

Gene	Expression	Region	Density	TF	TF Pattern
PRDM1	1124	Region_1	1114	PRDM1	1124
PRDM1	1124	Region_123	1114	IRF4	1124
IRF4	1124	Region_67	1114	PRDM1	1124
IRF4	1124	Region_90	1234	BACH2	4431

TABLE 6.1 – Example of the query output

6.5 Compatibility table to assign sign to relations

To regulate a gene, a TF must be expressed and be able to bind in an accessible chromatin area. A TF can either positively regulate or inhibit a gene expression. In the first case (activation), an open chromatin region and an expressed TF will induce the transcription of the gene and in the second case (inhibition) the gene will not be expressed. Based on these basic principles, we devised a compatibility table for each cell population.

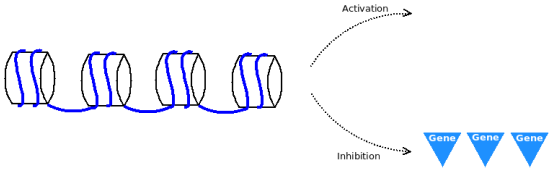
The motivation for this compatibility table is (1) to discard the TF-gene candidate relations that are not consistent with the biological knowledge of how regulation works, and (2) to infer a regulation sign (i.e. activation or inhibition) for the consistent candidates relations.

RD	TF	Gene	Sign
1	1	1	+
⋮	⋮	⋮	⋮
4	4	4	+

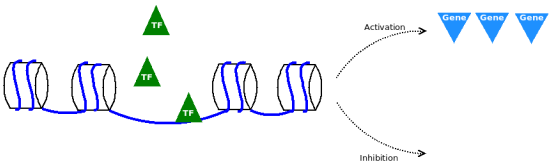
(a) Extraction of the regulation sign attribution table (+): *RD= Read density pattern, TF= TF expression pattern, Gene=Gene expression Pattern, Sign: potential sign of the regulation*

RD	TF	Gene	Sign
1	1	4	-
⋮	⋮	⋮	⋮
4	4	1	-

(b) Extraction of the regulation sign attribution table (-): *RD= Read density pattern, TF= TF expression pattern, Gene=Gene expression Pattern, Sign: potential sign of the regulation*



(c) Closed chromatin and non-expressed TF: *Activation: there is a potential lack of gene expression. Inhibition: the gene expression could be high.*



(d) Opened chromatin and expressed TF:*Activation: there is a potential high gene expression. Inhibition: the gene could be absent.*

FIGURE 6.10 – Illustration of the regulation sign attribution table.

FIGURE 6.10 presents a graphical representation of the compatibility principle behind assignment tables. When the chromatin is closed and the TF not expressed (First line of FIGURE 6.10A and FIGURE 6.10B): if there is no gene expression then the TF is more likely to act as an activator and if there is a high gene expression then the TF is more likely to act as an inhibitor. When the chromatin is opened and the TF expressed (last line of FIGURE 6.10A and FIGURE 6.10B): if there is a gene expression then the TF is more likely to act as an activator and if the gene expression is low or null, then the TF is more likely to be an inhibitor.

The compatibility table must follow the following principles:

— the maximum effects on the gene expressions are obtained when the TF is at its highest expression level.

— The more accessible the region, the higher is the impact of the TF on the gene: for an activation this implies that if the TF is highly expressed so must be the gene, for an inhibition the higher the TF, the lower the gene's expression.

— If the region looses its accessibility, the weight of the TF lowers: for an activation we need higher TF level to gain a similar gene level. For an inhibition, we would need higher TF level to lower the gene expression.
— The closeness of a Region can produce an effect comparable to an inhibition for activator TF by reducing its impact.

These principles can be seen in FIGURE 6.11: The window of activation (respectively inhibition) glides to lower (higher) gene expression as the TF expression decreases. The same behaviour is true when the region accessibility decreases. The only exception to this is when the value of one or more item is set to 5, which does not mean that the expression is higher but that it is constant across all studied populations.

For each digit of the three patterns (gene, TF and region), we check on the compatibility table if the relation is compatible with an activation or an inhibition. If it is the case, a score from 1 to 2 depending on the confidence ("-" and "+" award a score of 1; "- -" and "++" award a score of 2) is given to the pattern point and we move to the next. The sum all of points must be superior to a fixed threshold to award the relation a sign, either + or - depending of the direction. For example this threshold is fixed to 7 for a 4 digit pattern, allowing at most one point to be a little less compatible (lower confidence, see Figure 6.11).

After using the regulation sign attribution table for filtering we obtained a result in form of a quadruple between the gene, the neighbor region, the TF with a binding site in the region and the potential regulation on the gene such as presented in TABLE 6.2. These results represent a potential TF impact on gene expression and not an actual biological impact, which would need to be experimentally validated.

Gene	Expression	Region	Density	TF	TF Pattern	Regulation
PRDM1	1124	Region_1	1114	PRDM1	1124	+
PRDM1	1124	Region_123	1114	IRF4	1124	+
IRF4	1124	Region_67	1114	PRDM1	1124	+
ECH1	1124	Region_7569	4431	BACH2	4431	-

TABLE 6.2 – Regulation sign attribution: examples of the sign attribution output

TF \gene	1	2	3	4	5
1	++	+			
2	+	++	+		
3		+	++	+	+
4			+	++	+
5			+	+	++

(a) Activation, Region = 5

TF gene	1	2	3	4	5
1			-	- -	-
2		-	- -	-	-
3	-	- -	-		
4	- -	-			
5	-	-			- -

(b) Inhibition, Region = 5

TF gene	1	2	3	4	5
1	++	+			
2	+	++	+		
3		+	++	+	+
4			+	++	+
5			+	+	++

(c) Activation, Region = 4

TF gene	1	2	3	4	5
1			-	- -	-
2		-	- -	-	-
3	-	- -	-		
4	- -	-			
5	-	-			- -

(d) Inhibition, Region = 4

TF gene	1	2	3	4	5
1	++	+			
2	++	+			+
3	+	++	+		+
4		+	++	+	
5		+	+		++

(e) Activation, Region = 3

TF gene	1	2	3	4	5
1			-	- -	
2			-	- -	-
3		-	- -	-	-
4	-	- -	-		
5		-	-		- -

(f) Inhibition, Region = 3

TF gene	1	2	3	4	5
1	++				+
2	++	+			+
3	++	+			
4	+	++	+		
5	+	+			+

(g) Activation, Region = 2

TF gene	1	2	3	4	5
1				- -	
2			-	- -	
3			-	- -	-
4		-	- -	-	-
5			-	-	-

(h) Inhibition, Region = 2

TF gene	1	2	3	4	5
1	++				
2	++				
3	++	+			
4	++	+			
5	++	+			

(i) Activation, Region = 1

TF gene	1	2	3	4	5
1				- -	
2				- -	
3			-	- -	
4			-	- -	
5			-	- -	

(j) Inhibition, Region = 1

FIGURE 6.11 – Sign attribution table, divided by region pattern value. Cells with "++" or "- -" have a higher confidence than the "+" or "-" cells.

6.6 Automation

In close collaboration with Guillaume Collet, we proposed an automated version of the pipeline as a package available on Gitlab and optimized to run on the Genouest cluster [2].

The pipeline needs as input a coma or tabulation separated file with a list of differently expressed genes and a second with differently activated regulatory regions, it also needs two bed files with the coordinates of genes and regulatory regions. The TF binding sites localization are already implemented (extracted from *Regulatory Circuits*) but can be given by the user as bed files.

The automated version uses a combination of Conda and Snakemake to automatize the succession of the different scripts. Starting at the last step of the pipeline, the pipeline verifies if the resulting file is computed, if not it looks for the files necessary to run this step, if they do not exist the pipeline then move to the previous steps and so on until it can either run the step or runs out of steps and gives out an error. The steps represented in FIGURE6.12 are themselves implemented using Python and the Panda library to efficiently navigate data-sets and Bedtools for the intersection of the TF binding sites and region localisation. Once all the files are transformed into Turtle (ttl) files they are integrated into a local docker of Virtuoso and queried to find TF-region-gene relations. They are then run through the compatibility table and given signed TF-gene relations as output.

2. https://www.genouest.org/2017/03/02/cluster/

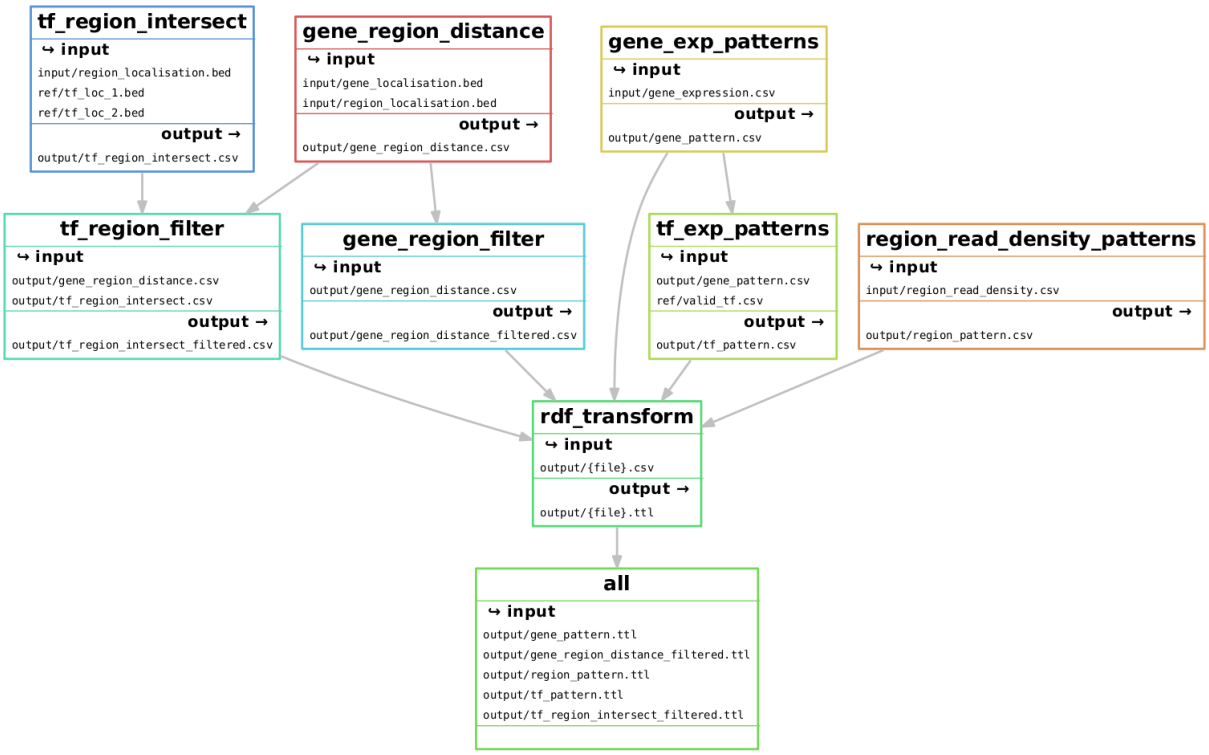


FIGURE 6.12 – Automation diagram up to integration: if the resulting file of a box is find the box is skip, else it is computed to get that resulting file.

For a set of 26,802 genes and 58,449 regulatory regions, on 4 different cell populations, the pipeline takes 59 minutes and 36 seconds to be completed and resulted in 5,635,099 relations TF-region-Gene and 314,965 TF-genes unique relations. Those relations are the same as the ones obtain by running the steps of the pipeline independently.

6.7 Comparison between *Regulatory Circuits* workflow and our pipeline

Our pipeline and *Regulatory Circuits* workflow both take similar data as input: activities of the regions and the genes (or approximation of the transcript activity) and regions localisation. We also have similar steps: closeness between the regions and the genes, finding the binding sites of the TF in our regions. All theses steps are listed in TABLE 6.3, the main differences are:

— We use the activity of the TF (extracted from its coding gene activity), they do not use this information.
— They approximate the gene to the transcript and the activity of the transcripts to its promoters, we directly use the gene activity (if available).
— *Regulatory Circuits* use a score in which each step that must be superior to 0, while we check the consistency of the activities of the elements of the relation.
— We also give networks by patterns whereas *Regulatory Circuits* gives tissue-specific networks.
— We produce signed networks.

Steps	***Regulatory Circuits***	**Us**
Activities transformations (Regions)	Ranks(promoter) & Rank(enhancer)	Pattern(Region)
Activities transformations (genes/transcripts)	Rank(Transcript) = max(Ranks(Promoter))	Pattern(Gene)
Activities transformations (TF)	$\emptyset$	extract TFs from Genes' pattern list
Link TF-Region	max(Confidence BS in region)	at least 1 BS in region
Link Region-Gene/Transcript	Distance $<$ |500kb| and Weighted	Distance $<$ |500kb|
Link transcript-Gene	correspondence file	$\emptyset$
Networks non-filtered	**1 overall network**	**1 overall network**
Filtering	score $>$ 0	compatibility table
Final Network	**By tissues** (Scored & Unsigned)	**By patterns** (Unscored & Signed)

TABLE 6.3 – Comparison between the steps of *Regulatory Circuits* workflow and our pipeline. The main differences are the use of Patterns instead of Ranks and the filtering steps. But also in the produced networks: unsigned and score tissue-specific in *Regulatory Circuits* and signed but unscored pattern-specific in our case.

6.8 Validation

For validating the pipeline, we ran it with *Regulatory Circuits* data, approximating the expression for the genes by that of their promoters. We selected 4 combinations of 4 tissues (2 sets of 4 biologically-similar tissues and 2 sets of 4 dissimilar tissues) representing 12 tissues from *Regulatory Circuits* (see TABLE 6.4), on which we had RNA-seq information from Roadmap Epigenomic to use as comparison for the topology of the network (as described in SECTION 3.4.3), for a total of 23 samples. On these samples, we ran both our pipeline and *Regulatory Circuits*'s using the bash implementation (see SECTION 3.4) as it was the only one allowing us to re-compute ranks and scores for smaller sets.

Type of sub-set	Tissue 1	Tissue 2	Tissue 3	Tissue 4
Similar (1)	B lymphoblastoid cell line	CD4+ T cells	CD8+ T cells	peripheral blood mononuclear cell
Similar (2)	colon adult	colon fetal	small intestine adult	small intestine fetal
Dissimilar (1)	CD34+ stem cells adult	brain fetal	epitheloid cancer cell line	pancreas adult
Dissimilar (2)	CD4+ T cells	brain fetal	colon adult	epitheloid cancer cell line

TABLE 6.4 – Composition of the tissues to form 4 sub-sets of regulatory circuits: 2 composed of biologically-similar tissues and 2 composed of dissimilar tissues. CD4+ T cells and Colon adult tissues both appear in similar and dissimilar sub-sets to be able to compare them.

TABLE 6.5 presents the number of relations in the resulting networks of the 4 set of tissues. Just after the query and before running the compatibility table, we can see that we had the same number of relations (3,005,934 TF-region-Gene or 1,869,854 TF-Gene), this is due to the fact that we have the exact same information on the TF binding site and regions for the four sets. We used the files given by *Regulatory Circuits* for all theses steps. The differences start after the compatibility table where the inconsistent relations are filtered.

In *Regulatory Circuits* the number of potential TF-genes relations is of 3,260,087. This number was found by making a union of 1) all the unique TF-gene relations appearing in at least one of the 394 tissue-specific networks (3,246,008) and 2) all the possible relations in *Regulatory Circuits* ignoring the scores, based on the provided intermediary files of TF-promoter, TF-enhancer, promoter-transcript and enhancer-transcript relations (2,060,960). All the TF-gene relations found by our pipeline are included in the potential *Regulatory Circuits* relations, meaning that our method does

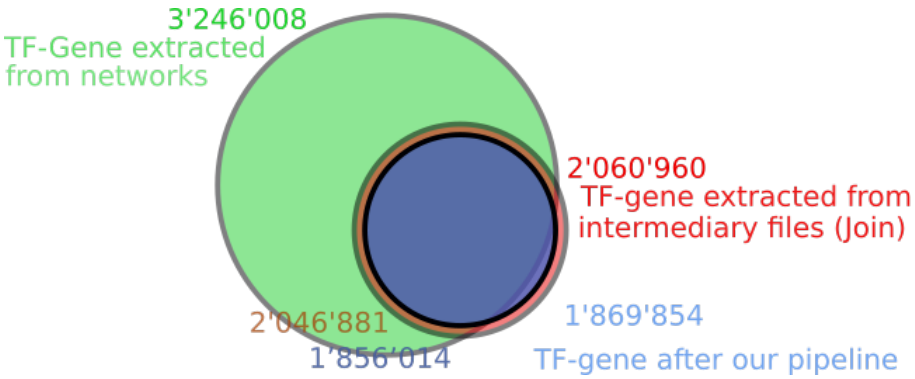


FIGURE 6.13 – Number of unique TF-genes relations found in results networks (in green) (1), number of relations found while following the files (red) (2) and number of relations found while using our pipeline (blue) (3). The number of relations in blue is lower than the relation in red because we exclude TF with binding score of 0 in our pipeline. All relation we compute (3) s either included in (1) or (2), 13,840 are not included in (1) and only in (2).

not create irrelevant relations. But 13,840 were only found in the list of potential relations (2) we hypothesize that those relations are scored to 0 in the *Regulatory Circuits* networks. We also found fewer relations than (2) because we excluded all relations where the confidence score of the TF is 0. We can see the intersection of each set of relation in FIGURE 6.13. We do have fewer relations than the maximum proposed by *Regulatory Circuits*, in part due to the fact that we exclude all TF with a 0 score for their binding site while processing the files, so we only kept 596 TF out of the 643 in the original data set.

Sub-set	**Nb relations before Table**	**Nb relations After Table**	**Nb relations Unique**	**Nb relations "+"**	**Nb relations "-"**
Similar 1	3,005,934	219,495	164,251	114,624	49,627
Similar 2	3,005,934	237,487	178,514	154,276	24,238
Dissimilar 1	3,005,934	165,804	125,451	79,359	46,092
Dissimilar 2	3,005,934	165,597	126,145	88,609	37,536

TABLE 6.5 – Number of relations by network after the pipeline, on the 4 sub-sets of *Regulatory Circuits*.

As seen in TABLE 6.5 the networks computed on dissimilar sets of tissues are slightly smaller and contain (relatively to their number of relations) more inhibitions relations than the ones computed with similar subsets. The lower number of relations

can be explained by the lack of consistency in TF-gene regulations across different tissues.

In FIGURE 6.14, we look at the percentage of genes we are able to find in our network regarding the RNA-seq expression level given as validation in *Regulatory Circuits*. We can see that for the networks computed using our pipeline, we recover sightly fewer genes from the 10% the most expressed (-4% for similar and -6% for dissimilar) and for the middle 10% (-3% for similar and -4% for dissimilar). But we do gain a lot of least expressed genes: from 17% in the original networks, to 36.75% with similar cells-types and 52% when running our pipeline with dissimilar sub-sets.

The lack of some highly expressed genes can be explained by the number of relations we obtain which is inferior to the number found in the original graphs. We do keep a lot more of least expressed genes because - contrary to *Regulatory Circuits* original workflow - we do not favor inductions and keep inhibitions in the result networks. We can also recover them if they are activated in another cell type of the subset. This recovery of more of the low expressed genes is a expected output of our pipeline as we do not exclude based on the scores.

TABLE 6.6 presents the same type of information about the RNA-seq percentage found, but focuses on the two tissues used in both similar and dissimilar runs of our pipeline. We can see the same trend: similar sets allow for better coverage of top and middle part of the RNA-seq but dissimilar sets recover more of the least expressed genes.

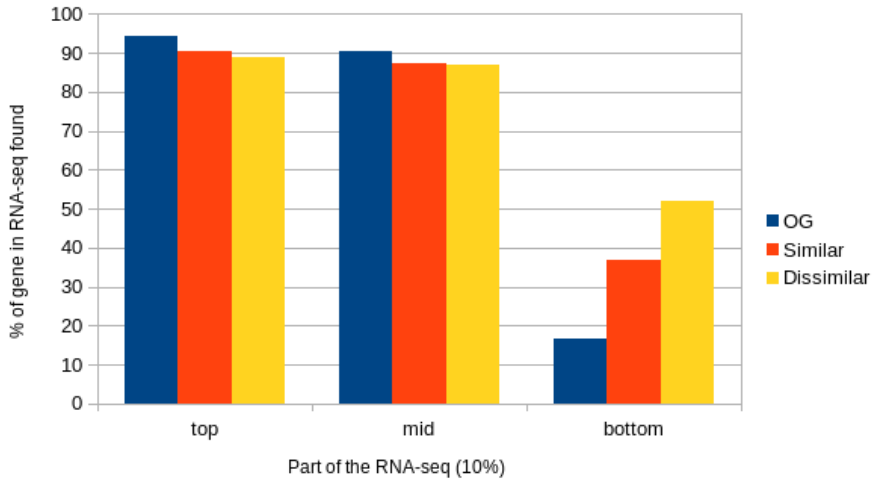


FIGURE 6.14 – Percentage of genes from the RNA-seq related to the networks found in the resulting networks. The RNA-seq genes are separated in three categories: the top 10% most expressed, the middle 10% and the 10% least expressed. In blue the original networks as computed by *Regulatory Circuits*, in red the network computed with our pipeline on similar data-set and in yellow networks computed with our pipeline but with dissimilar cell-types.

Sub-set	Network	Top 10%	Middle 10%	Bottom 10%
Similar	**colon_adult**	91	89	32
Dissimilar	**colon_adult**	89	87	35
Similar	**CD4**	91	86	41
Dissimilar	**CD4**	89	84	66

TABLE 6.6 – Percentage of genes from the RNA-seq related to the networks found in the resulting networks. Focus on Colon Adult and CD4+ tissues both found in either similar or dissimilar sub-sets.

In a second time, we looked at the signed relations. As this information is not contained in *Regulatory Circuits* we looked at the two major databases containing signed regulatory relations to confirm the signs: Trrust [HAN et al., 2018] and Signor [LICATA et al., 2020]. Trrust and Signor both contain signed relations that have been found through literature. In Trrust some relations are unsigned but this still adds an information: the relation between the TF and the gene have been found in literature. Trrust also has relations that are both signed as activation and inhibition (and unsigned at the same time on some examples).

In TABLE 6.7 we compiled the relations in our networks that are found in either Trrust or Signor. In average we found 0.22% of the relations of a set in Trrust and 0.04% in Signor. In Trrust 45% are unsigned, 39% signed in the same direction and 16% signed differently as the database. In Signor 76% are signed the same way and 23% in opposite direction. The relations found in common between Trrust and Signor in our networks are signed the same way in most cases and in 72% signed the same way as in our networks.

Graph	**Trrust**				**Signor**				**Trrust ∩ Signor**			
	Total	True	False	Unknown	Total	True	False	Unknown	Total	Different	True	False
Similar1	432	164	65	203	76	54	21	0	51	9	33	9
Similar2	357	156	37	164	54	45	9	0	39	5	31	3
DiSimilar1	293	100	54	139	56	42	14	0	39	4	28	7
DiSimilar2	264	113	50	101	58	44	14	0	37	2	26	7

TABLE 6.7 – Relations found in Trrust and Signor and coherence of signs. True: number of relations with the same sign as the database, False: relations with different sign than the database, Unknown: relations non signed or signed + and - in the database. For the union of Trrust and Signor: different: relations signed differently in the two databases, True: relation signed the same as both databases and False: relations signed differently than the databases.

For the relations generated by our pipeline and found at least one of Signor or Trrust, the sign we predicted is consistent with the databases in two third of the time. It is important to note that Trrust and Signor relations are not necessary found in the same tissues as the one we use and that can explain some of the differences in signs.

6.9 Conclusion

In this chapter we presented a new design for regulatory network inference. Our pipeline addresses some of the methodology concerns presented in *Regulatory Circuits*: (1) the lack of consideration for the TFs expression, (2) the discretization of the expression, (3) the favoritism of activation regulation and (4) the lack of signed relations. To solve theses issues (1) we added the TFs expression in the pipeline by looking at the expression of the gene that code for them, (2) we choosed to use patterns of expression which cluster genes or regions of similar expression direction under the same pattern. We computed an overall networks and then checked the consistency of the re-

lations with the biological background, which allowed us to sign the relations (4) and to not discriminate the inhibition (3). We also provide an automated version of our pipeline to facilitate its reuse.

Our pipeline gives is consistent with the expression of more genes than *Regulatory Circuits* workflow. We were able to explain the regulation of poorly expressed genes by looking into the regulations of inhibition. We also confirmed that all found relations between a TF and a Gene are already existing in the realm of potential relations of *Regulatory Circuits*.

CHAPITRE 7

APPLICATION TO B CELLS AND INTERPRETATION

In this chapter, we apply the pipeline presented in CHAPTER 6 onto a specific set of four closely-related cell types found in the B cell differentiation: naive B cells, memory IgG. and IgM. and plasma blast. We then extract the main TFs of the regulation to further biologically experiment and find key regulator of this differentiation.

In SECTION 7.1 we overview the application of our pipeline on this data-set, from the type of input data to the data integrated in an end-point and the resulting networks before and after filtering. As the genes are grouped into patterns we then (SECTION 7.2) look into the patterns interactions which each other as a higher level of regulation. In SECTION 7.3 we look for the master candidates of the regulation: the TFs with an impact on a substantial part of a pattern and specific enough to regulate the differentiation in only one direction. We look at the candidate found an their consistency with the literature.

7.1 Introduction

The pipeline presented in the previous chapter is meant to run on small data-set of closely related cell populations. We are particularly interested in deciphering the transcriptional regulatory network changes sustaining a hematological malignancy arising form B lymphocytes, follicular lymphoma (FL). To better understand the FL regulatory networks, we need to identify these networks in a physiological situation. Therefore, a first step was to run the pipeline on data from normal B cell differentiation in order to have a baseline of comparison to see the perturbations of the network for the FL. One of the advantages of the NBC differentiation is that it is a biological process better

understood than the FL mutations, with some regulators already known and expected. We thus had a basis for comparing the results of the pipeline to the bibliography.

In this chapter we apply the pipeline to four B cells populations and look at the resulting networks. We also describe methods to filter the resulting networks in order to extract TF which could be key regulators of the B cell differentiation. The goal of the network inference and limitation of the TF space to explore is to be able to give a list of TF of interest to the biologists we are working with, in order to further biologically experiment and confirm the regulatory impact of the said TFs.

7.2 Application of the pipeline

7.2.1 Input data

In this study, we used four distinct populations: NBC, MBC IgM, MBC IgG and PB for which we had gene expression data (RNA-seq) and epigenetic data about chromatin accessibility (ATAC-seq) that can be used to determine potential regulatory regions. Some background knowledge on this differentiation is presented in FIGURE 7.1 ,in this specific case the four populations are sequential: NBC is the first population and PB is the last, but MBC can be either a transitional state or a final one. Three main TF are highlighted in the bibliography at different steps of the differentiation: an inhibitor, BACH2, and two activators, IRF4 and PRDM1.

The initial data for this project were the genes names and coordinates of 29.261 identified genes in our populations. Their expression values were obtained as the normalized mean of three samples of RNA-seq data in each of the four populations. NBC, IgG and IgM memory B cells have been purified from blood samples and plasma cells have been produced *in vitro* and are close to extra-follicular PB. The differential expression analysis was run using DESeq2 R package: we found 14'921 genes that did not pass our expression threshold and 3'591 genes of constant expression in all four populations.

For these four populations we also had the information of read density in ATAC-seq, which lead to the identification of large sets of regions of open chromatin area. We had 35.078 ATAC regions each with specific names and coordinates.

As described in CHAPTER 6, the list of TF we look at is extracted from the *Regu-*

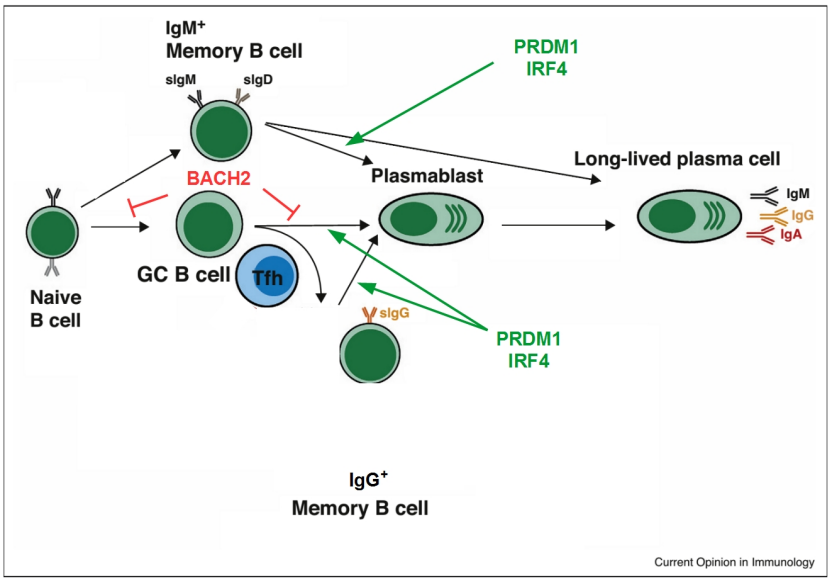


FIGURE 7.1 – NBC differentiation (simplification): Pathways for the generation of human B-cell subsets. Naive B cells can yield IgM-only or Ig class switched memory B cells in a GC dependent manner, as well as plasma cells. MBC can also differentiate into PB, although through distinct mechanisms. [PHAN et TANGYE, 2017]

latory Circuits initiative, their activity is extracted from the list of gene we looked at or put to 0 if not expressed in our data-set and their binding sites are filtered within our regions.

7.2.2 Integration

Once the data are pre-processed through the pipeline - computation of the distance between region and genes, finding the binding sites of the TF present in the given regions and transformation of expressions into patterns - the data are integrated using AskOmics. The database structure is a Triplestore as such interaction between two entities are described as triples.

With the entry data we obtain 109 distinct Patterns of 4 digits representing in order: NBC, MBC IgM, MBC IgG and PB. The patterns are comprising from 1 (1412 and 2414) to 1,418 (4441) genes, while the 0000 and 5555 patterns represent 14,921 and 3,591 genes, respectively - those two patterns were put aside the during the interpretation of the pipeline as they bore less information on the differentiation. 18 patterns are composed of more than 100 genes as presented in FIGURE 7.2. Included within the

genes but also treated as a different entity, we also have 593 TF out of which 326 had a pattern of 0000 hence would not impact the networks as we compute them.

In addition to the patterns, we computed 3,460,101 relations of binding sites inclusion between a TF and a region and 496,282 relations under 500kb between our ATAC regions and the genes.

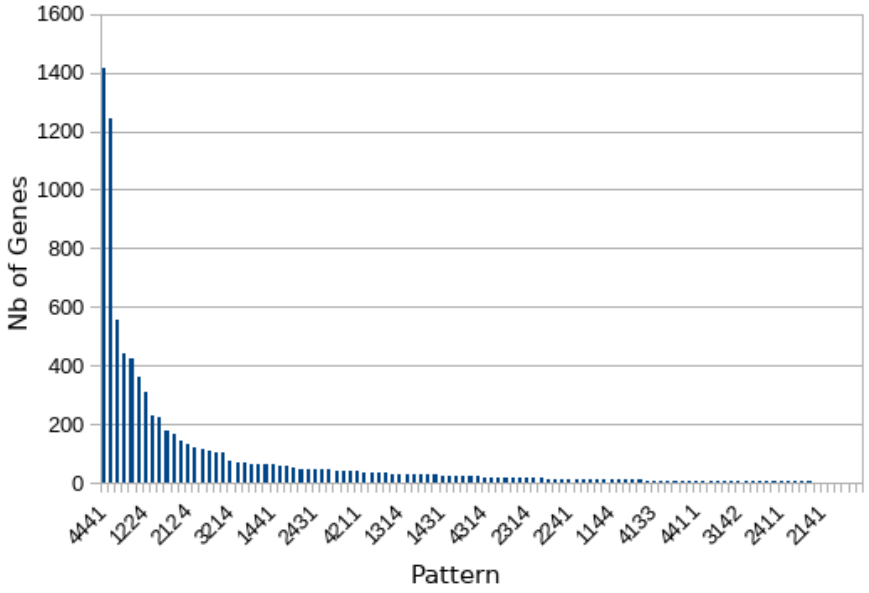


FIGURE 7.2 – The different Patterns present in the pipeline: 107 patterns (0000 and 5555 not represented), with 18 composed of more than 100 genes and 33 of less than 10 genes

TABLE 7.1 presents the different classes integrated into the AskOmics database, with their number of entities and their attributes. Not all attributes presented in TABLE 7.1 have been used to run the pipeline but were integrated in case we wanted to allow finer-exploration of the data. For example: limitation of the distance of a region from a gene, only looking for one chromosome, etc...

Class	Number of entities	Data types
Gene	29 261	Start, End, Duplicates, Transcripts, Chromosome, Strand, Expression Pattern
ID ATAC	35 078	Start, End, Chromosom, Density Pattern
ID ATAC Closest	496 282	Distance to gene, Neighbor Gene, ID ATAC
TF	593	Expression Pattern
TF inclusion ATAC	3.460.101	ID ATAC, TF

TABLE 7.1 – Entities integrated in the AskOmics database: the different classes and their data types integrated in AskOmics, and their number of elements.

As presented in TABLE 7.2 the total data-set integrated represents 14,675,450 and 6,552,236 distinct entities.

Triples	14 675 450
Entities	6 552 236
Classes	5

TABLE 7.2 – Data in AskOmics database: Number of entities and relations successfully integrated in the database.

7.2.3 Networks extraction and filtering

Once integrated the data-set is queried to retrieve all relations respecting the set conditions: all interactions involving a TF with a binding site in a region closer than 500 kb of a given gene. Taking all patterns into account excepted 0000, this query outputs 5,635,099 relations. We can then filter the result of the query with the compatibility table resulting to 612,633 TF-region-gene relations with a score above the compatibility threshold. We can then reduce the number of relations by merging TF-gene relations that happen through different regions: we then obtain 314,965 unique TF-gene relations consistent with the fixed threshold of compatibility.

As we are more focused in the regulation of the global differentiation than on direct regulation between a TF an a Gene, we choose to compile the information given in the network as a global file giving all the TF and the number of their target in each pattern as the direction of the regulation. A sub-part of this file can be found in TABLE 7.3. We pass from 314,965 unique TF-genes relations in the previous step, to 7,465 unique relations between TF and patterns, as several TF-genes relations are fuse in the same

TF-pattern relation. The global table contain 64,637 potential relations of which 3,050 are activation (+), 3,933 inhibition (-), 482 are found either as activation or inhibition for the same TF-pattern couple (+/-). The remaining 57,172 relations are not found in our data and are put to 0.

Pattern	4441			4431			4421			4414		
TF	**nb Gene**	**Percent**	**Regulation**	**nb Gene**	**Percent**	**Regulation**	**nb Gene**	**Percent**	**Regulation**	**nb Gene**	**Percent**	**Regulation**
CEBPA	28	1	+/-	12	2	+/-	0	0	∅	0	0	∅
CEBPB	29	2	-	0	0	∅	0	0	∅	1	33	-
CEBPG	499	35	+	148	33	+	0	0	∅	0	0	∅
CENPB	0	0	∅	1	0	+	0	0	∅	0	0	∅
CLOCK	3	0	+	0	0	∅	0	0	∅	0	0	∅
CREB3	665	46	-	35	7	-	30	43	-	0	0	∅
CREB3L1	0	0	∅	0	0	∅	0	0	∅	0	0	∅
CREB3L2	379	26	-	115	26	-	22	31	-	0	0	∅
CTCF	796	56	-	190	43	-	29	42	-	0	0	∅
CUX1	0	0	∅	86	19	+	9	13	+	0	0	∅

TABLE 7.3 – Sub-part of the general file of interaction between TF and the different patterns. Include the number of genes targeted by the TF in the pattern, the percentage of the pattern it represents and the direction of the regulation.

For example the output for the 1124 pattern gives: 7,375 TF-Region-Genes relations and 6,004 TF-genes relations for 519 distinct genes (out of the 557 existing genes in the patterns) and 145 TF (of which 12 are also genes member of the pattern), out of which 3,777 are activation (+) and 2,227 inhibition (-). This is represented in FIGURE 7.3 which creates an hyper-dense network not easily navigated.

7.3 Patterns interactions

At first, we can focus on a overall graph composed of the different patterns and how they interact with each others. This is done using TABLE7.3 and looking at the pattern of the different TF. An edge between two pattern is defined as the interaction between at least a TF from a pattern (1) to a target gene in an other pattern (2): creating the relation pattern (1) regulating pattern (2). The size of the edge is defined by the number of TFs regulating the target pattern. The graphs of Pattern can highlight the combinatory aspect of the regulation and more precisely the potential cascade of regulation as TF are themselves regulated by other TF.

This first step was to extract all “pattern-on-pattern” relations, as we can see in FIGURE 7.4 this lead to a high number of edges between the patterns and a very

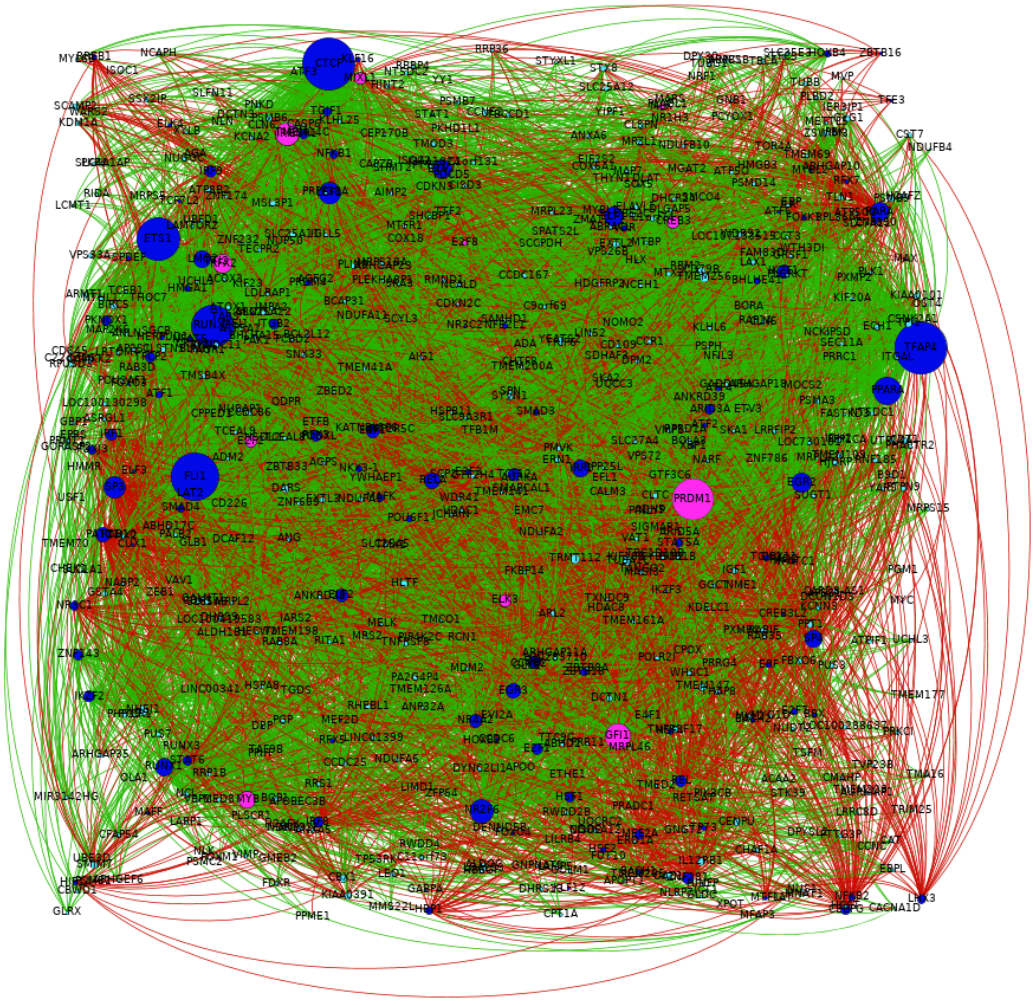


FIGURE 7.3 – Visual representation of the network for the pattern 1124. Representing 6,004 edges, 652 nodes (507 genes (dark blue), 133 TF (Turquoise) and 12 both TF and Genes (pink)), the size of the node represent the number of connections they have.

dense graph. The pattern 0000 has been represented but not connected as it is filtered during the compatibility table.

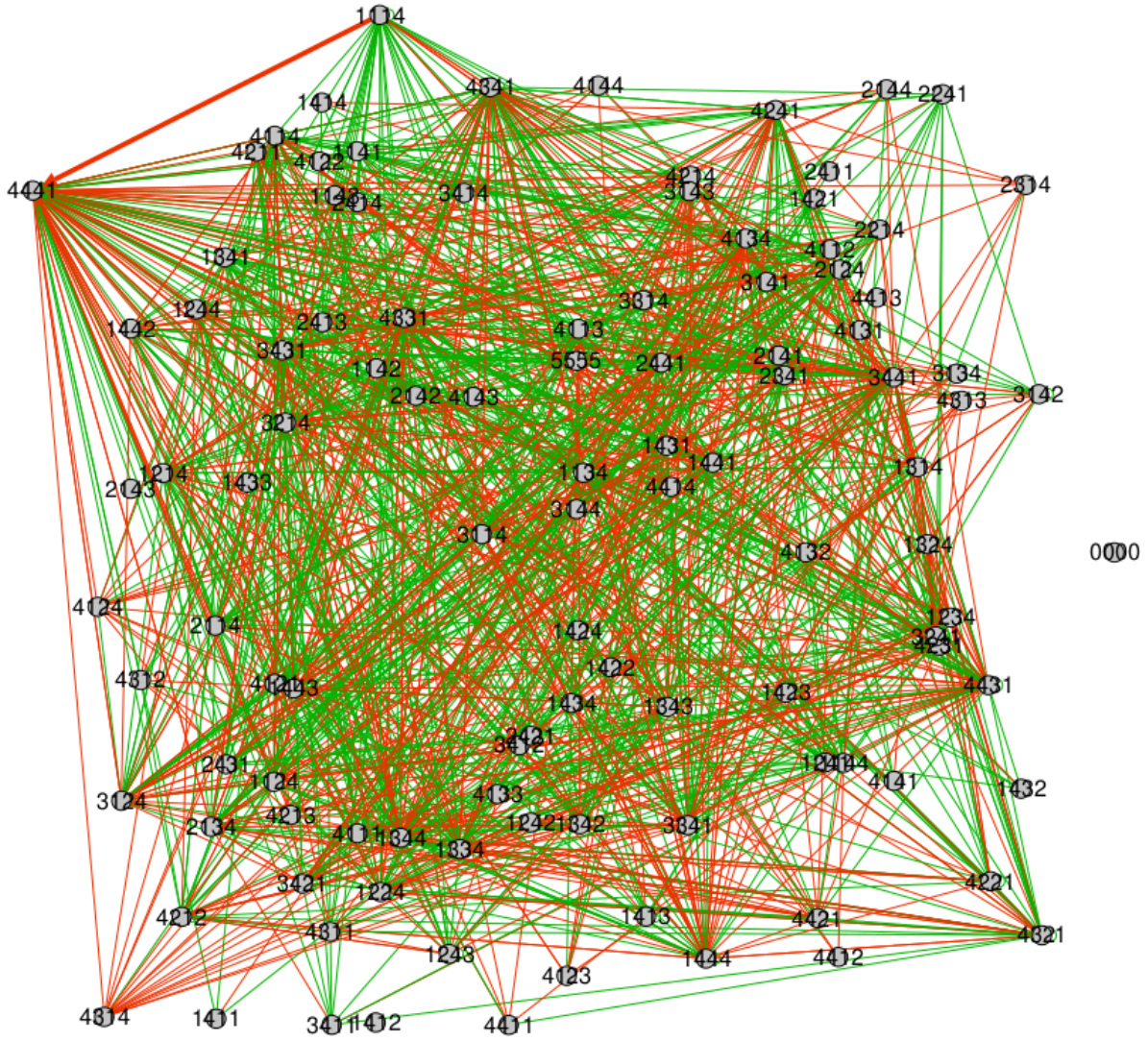


FIGURE 7.4 – Interactions between all the patterns. Note that 0000 is not connected. In green arrow are represented the activation (regulation +) and in red the inhibitions (-).

As the previous graph is too dense to look at and extract relevant information, we then focused on the 18 patterns with more than 100 genes FIGURE 7.5. We limited ourselves to the interactions within the 18 patterns on which we added the main inferred TF. A TF was only added if it was part of the 5 regulators with the most targets in another pattern. We can see several auto-activation loops (1124, 4441, 1224, etc...)

and mutual regulations when two patterns are regulators of each other (4331-4321 and 4441-1124).

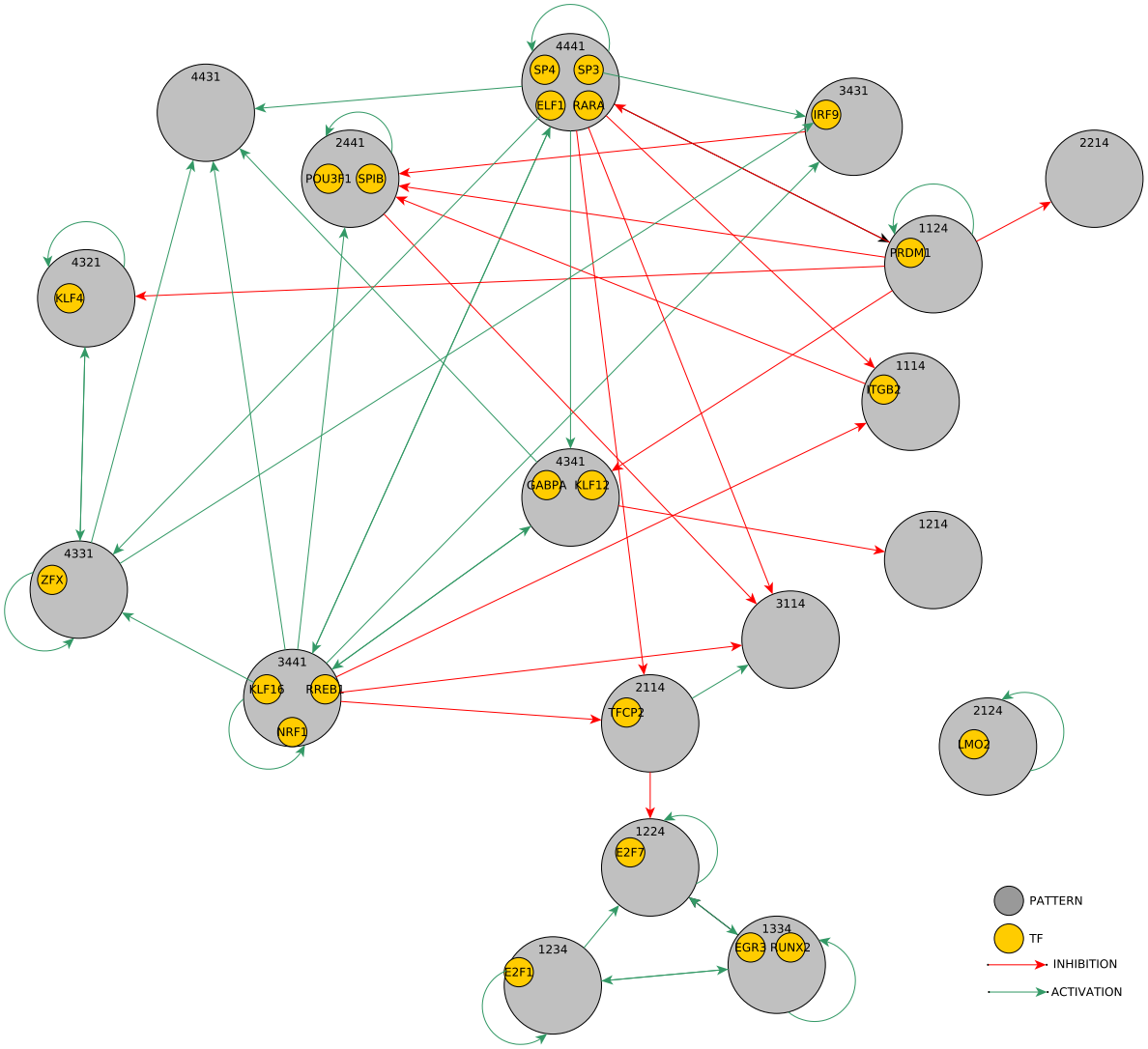


FIGURE 7.5 – Interactions between patterns, limited to the interactions between the 18 patterns with more than 100 genes. We also added the main TFs in each patterns: a TF was added if it was part of the 5 TF with the most targets in at least one pattern.

The last aspect considered is the closeness of some patterns that have either the same direction (ex: 1124 and 1224 or 1134 and 4441 and 4431) or completely opposite direction (ex: 1124 and 4431). The closeness was defined as a distance of maximum 1 between both pattern. If we focus on the 18 patterns with more than 100 genes we find two main directions: pattern going up from low expression in NBC to high expression

in PB and patterns going down. This is shown in FIGURE 7.6 were we can see that all patterns in the 18 are related to at least one other.

We can see by comparing FIGURE 7.5 and FIGURE 7.6 - in which the patterns are placed in the same order - that closes patterns of same directions are activators of each others (top-left of the figures and triangle at the bottom, example: 4431 and 4441). While patterns opposite direction are often inhibitor of each other (Right to left of the figures, example: 3441 and 2114).

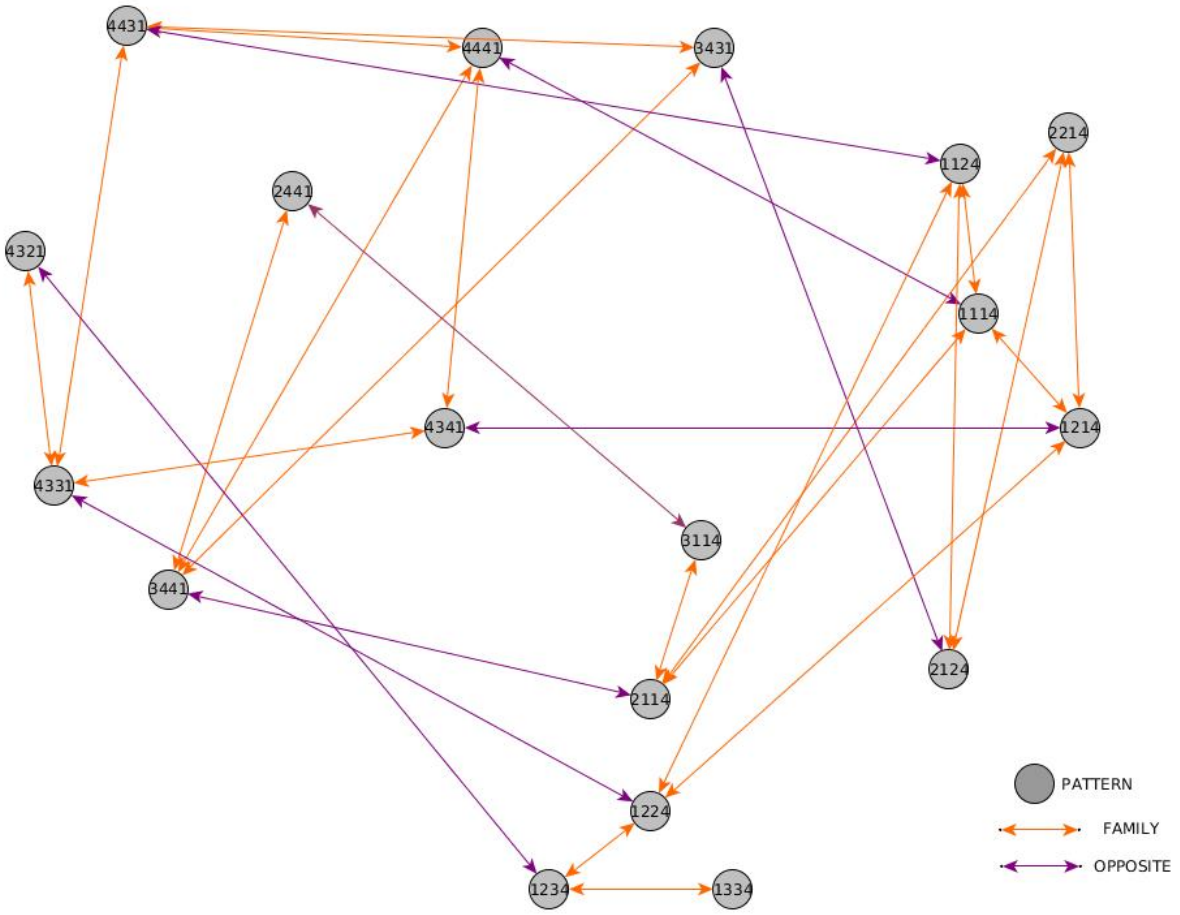


FIGURE 7.6 – Closeness of Pattern: on the left pattern that go down, on the right pattern going up

7.4 Finding master candidates of the regulation

In our study, a good regulator candidate is a TF that we infer to have an impact on a substantial part of a profile representing a specific part of the global network. Those two criteria are based on the need to experimentally validate the TF after the computation to verify if they indeed have a biological action on the regulatory network. The more targets the TF have the more likely it will be to identify its function in experiments, particularly if they have a specific expression profile. Since we also infer a sign of the relation, the expected behaviour of the TF will be even more defined and easy to confirm.

In this section, we mainly focus on the network for the gene pattern 1124, as it is a pattern of genes which are known regulators of B cell differentiation, such as PRDM1 and IRF4.

7.4.1 Coverage

The first aspect we focus on is the coverage of a TF: it is the number of genes a TF has for targets in a specific Pattern. As the Pattern have various number of genes, we used the percentage of the pattern covered as the final value of the coverage. The higher the coverage, the higher is the number of genes of the patterns potentially regulated by the TF.

In FIGURE 7.7 we present a small toy sample of resulting network composed of 1 TF and 4 genes. In this example the TF cover 100% of the pattern2 (1 out of 1 genes) and 66% of pattern1 (2 out of 3 genes).

The network of the pattern 1124 is covered by 152 TF, of which 20 cover more than 50% of the pattern and only 2 more than 80%. The distribution of the TF coverage is shown in FIGURE 7.8: 73 TF cover less that 100 genes in the pattern. We define top5 (respectively top10) as the 5 (10) TF with the most target in a given pattern, it is the 5 TF with the higher coverage in said pattern.

TABLE 7.4 present the top5 TF in terms of coverage for the pattern 1124. We can see a higher pondering of inhibition than activation relations and we find one of the known TF: PRDM1.

TABLE 7.5 present the top10 regulators in coverage for 18 patterns (all with more

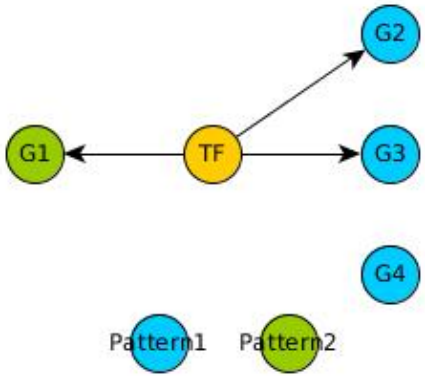


FIGURE 7.7 – Toy-example: in this example the TF covers 100% of Pattern2 and 66% of Pattern1

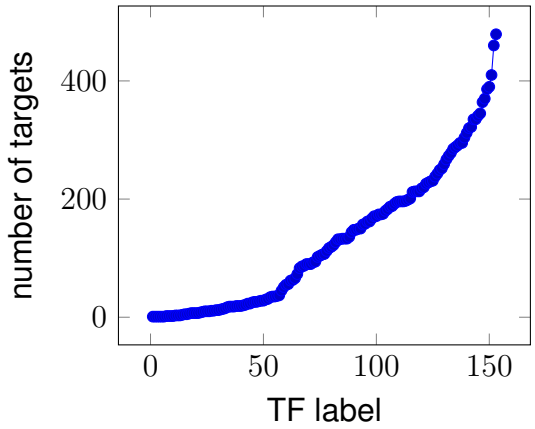


FIGURE 7.8 – Distribution of the number of target by TF on the pattern 1124 (557 Genes). The TF are labeled according to their number of targets.

Targeted genes	%	TF	Regulation
386	69	RARA	neg
390	70	ELF1	neg
410	73	PRDM1	pos
460	82	SP3	neg
479	85	SP4	neg

TABLE 7.4 – Focusing on Pattern 1124: Top 5 regulators by number of targeted genes and their regulation

than 100 genes). We can see some highly represented TF across all patterns, with high levels of coverage. For example SP4 is the Top1 regulator in 8 patterns and Top2 in 2. Some other notable examples are RARA (present 11 times), SP3 (10), ELF1 (10), RREB1 (10) SP1 (9) and KLF16 (9). As they have a large number of targets they do not appears as interesting for the biological experimentation: they seem ubiquitous and would not give a specific direction to the model to be able to follow their regulatory impact.

But the TABLE 7.5 also shows that in opposed pattern (ex: 1124 and 4431) the same TF have opposite regulatory roles: SP3 and SP4 have an inhibitory impact in 1124 and are activators in 4431. 15 out of the 18 patterns are covered at least at 90% with the combined actions of their top5 regulators, but are never totally explained only by looking at this top5.

Pattern	Total of Genes in Pattern	Covered by top5	in %	Top5 TF	Top6 to Top10 TF
3431	116	110	94.83	SP4 (+) KLF16 (+) SP3 (+) ZFX (+) ZNF219 (+)	ELF1 (+) RARA (+) RREB1 (+) NRF1 (+) PATZ1 (+)
2114	421	399	94.77	SP4 (-) KLF16 (-) SP3 (-) RREB1 (-) ELF1 (-)	EGR1 (-) RARA (-) ETV6 (-) ELF3 (-) PATZ1 (-)
1214	221	209	94.57	SP4 (-) SP3 (-) KLF7 (-) ELF1 (-) RARA (-)	GABPA (-) PATZ1 (-) KLF12 (-) ESR1 (+) SP1 (+/-)
3441	358	338	94.41	SP1 (+/-) SP4 (+) KLF16 (+) SP3(+) RREB1 (+)	ELF1 (+) RARA (+) PATZ1 (+) GABPA (+) TCF3 (+/-)
4321	102	96	94.12	KLF4 (+) ZFX (+) ZNF219 (+) TFAP4 (-) PRDM1 (-)	PPARA (-) EGR2 (-) ZBTB14 (+) RFX2 (-) ETV7 (-)
4441	1418	1329	93.72	SP4 (+) SP3 (+) KLF16 (+) SP1 (+/-) RREB1 (+)	PRDM1 (-) ELF1 (+) RARA (+) NRF1 (+) PATZ1 (+)
2441	110	102	92.73	KLF16 (+) RREB1 (+) ZNF143 (-) NRF1 (+) ITGB2 (-)	PRDM1 (-) SPIB (+) SP1 (+/-) YY1 (+) IRF9 (+)
2214	178	165	92.70	EGR1 (-) RARG (-) RORA (-) JDP2 (-) TCF7L1 (-)	RUNX1 (+) PRDM1 (-) TFAP4 (-) ETV7 (-) NR2F6 (-)
1124	557	516	92.64	SP4 (-) SP3 (-) PRDM1 (+) ELF1 (-) RARA (-)	TFAP4 (+) PATZ1 (-) RFX2 (+) RFX3 (-) EBF1 (-)
4431	439	406	92.48	SP1 (+/-) SP4 (+) SP3 (+) RREB1 (+) KLF16 (+)	ZFX (+) ELF1 (+) RARA (+) NRF1 (+) GABPA (+)
1114	1244	1148	92.28	SP4 (-) SP3 (-) RREB1 (-) KLF16 (-) KLF7 (-)	SP1 (+/-) ELF1 (-) RARA (-) NRF1 (-) PATZ1 (-)
4331	228	209	91.67	SP4 (+) KLF16 (+) SP3 (+) RREB1 (+) KLF7 (+)	KLF4 (+) ELF1 (+) ZFX (+) RARA (+) NRF1 (+)
1234	142	130	91.55	EGR3 (+) NR2F6 (+) RUNX2 (+) THRB (-) BHLHE41 (+)	HSF4 (+) E2F1 (+) ZNF143 (-) E2F7 (+) SP1 (+)
1334	105	96	91.43	RUNX2 (+) EGR3 (+) NR2F6 (+) SP1 (+) POU2F1 (+)	TCF3 (+) E2F1 (+) HSF4 (+) SPI1 (+) POU2F2 (+)
2124	131	118	90.08	CTCF (+) FLI1 (+) EGR1 (-) ETV6 (-) RORA (-)	JDP2 (-) LEF1 (+) LMO2 (+) SOX7 (-) ESR1 (-)
4341	168	151	89.88	SP4 (+) KLF16 (+) RREB1 (+) SP3 (+) KLF7 (+)	ELF1 (+) SPI1 (+/-) RARA (+) PRDM1 (-) SP1 (+/-)
3114	121	108	89.26	EGR1 (-) ETV6 (-) NFYA (+) RORA (-) JDP2 (-)	ZNF143 (+) POU3F1 (-) TFCP2 (+) RREB1 (-) RARA (-)
1224	307	259	84.36	EGR3 (+) RUNX2 (+) ETS1 (-) ZNF75A (-) E2F1 (+)	THRB (-) ZNF143 (-) E2F7 (+) ARID3A (+) TFCP2 (-)

TABLE 7.5 – List of the TOP TF for the patterns with more than 100 genes, list given in order of coverage.

As shown, using only the coverage is not sufficient to find good candidates for the biological experiment: it does not take into account the diversity of targets of the regulator.

7.4.2 Specificity

A second aspect we choose to focus on is the specificity, which is based on the number of targets of a TF that are from a specific pattern. A TF has a great specificity for a pattern if out of all its target a significant number come from this pattern. As for

the coverage, the specificity is calculated in percentage, the number of targets in the specific pattern on the total number of targets of the TF.

In FIGURE 7.9 we use the same toy-example as for the previous subsection. In this example the TF is specific at 66% of pattern1 (2 out of 3 targets) and at 33% of pattern2 (1 out of 3 targets).

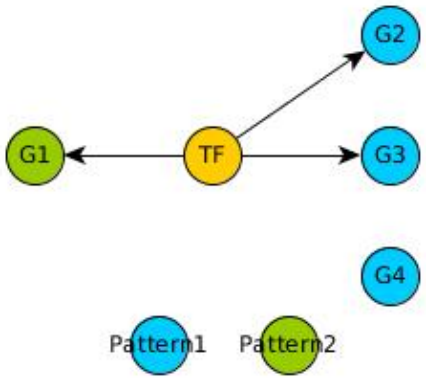


FIGURE 7.9 – Toy-example: in this example the TF is specific at 66% of Pattern1 and at 33% of Pattern2

TABLE 7.6 present the specificity of the TF PRDM1: it has targets in 35 patterns out of which only 3 represent more than 50% of all PRDM1 targets. PRDM1 is more specific to the pattern 4441 (34%) than of 1124 (13%) and 1114 (12%), 1124 and 1114 being very close patterns (low in all population except PB) and 4441 being the opposite of 1114.

TF	Total target	4441	1124	1114	3441	4431	4341	4331	4321	4221	2441	2214
PRDM1	3076	34	13	12	8	5	3	1	2	1	2	1

4421	4414	4411	4314	4311	4241	4231	4214	4211	3431	3421	3411	3341
1	0	0	0	0	0	0	0	0	0	0	0	1

3314	3241	3214	2431	2421	2414	2411	2341	2314	2241	1134
0	0	1	0	0	0	0	0	0	0	1

TABLE 7.6 – Specificity of PRDM1, in percentage of the target of the TF belonging to the pattern. Out of 35 patterns covered, 3 have a specificity over 10% and 28 with less than 2

As for the coverage, the specificity does not bring enough information by itself:

despite having a large number of its targets in a Pattern, a TF may have little influence on it if the pattern itself is very large.

7.4.3 Combination of coverage and specificity

The solution we chose to implement was to use a combination of the two previous criteria to extract some TF of interest. In this section we do not differentiate if the TF is an inhibitor or an activator in the pattern, we also kept unknown direction: if the TF is activator of one gene of the pattern but inhibitor of another of the same pattern. For a given pattern, a TF of interest is a TF whom specificity and coverage are both superior to their respective mean $+$ one standard deviation. The TF needs to have both its specificity and its coverage over the threshold. Then, they are two ways to find TF: either by focusing on the TF or on the pattern.

By focusing on the TFs, we obtain a list of patterns where the TF pass the threshold and might be of interest, i.e. patterns regulated by this specific TF. And by focusing on the Pattern, we obtain a list of TF that pass the threshold and might be regulators of this specific pattern.

In the following figures we reefer as "UP" the TF/Patterns for which the threshold is passed, the regulations are indifferently activation or inhibition.

FIGURE 7.10 focuses on the pattern 1124: for this pattern 10 TF pass the threshold for both coverage and specificity: CREB3, E2F2, ELK3, GFI1, IRF4, MYB, PPARA, PRDM1, RFX2 and TFAP4.

FIGURE 7.11 focuses on three TFs: IRF4, PRDM1 and BACH2. IRF4 and PRDM1 which seems to have an impact on 1134 and 1124, two related patterns and BACH2 on 3421. IRF4 and BACH2 were not TF that were high on the coverage list, and they did not appear in the top10 TF of their respective target pattern.

In FIGURE 7.12A we look at the number of TF we extract from the data. For 24 patterns we highlight no specific regulator and for 9 only one TF appears to pass both threshold, finally 12 patterns have 10 or more regulators of interest. As shown in FIGURE 7.12B, out of the 237 TF in the resulting networks, 91 do not pass the threshold for any patterns, 31 pass in only 1 pattern and 21 are up in 10 or more patterns.

The TF passing the threshold in more than 10 patterns can be less interesting that some passing in less: they are at higher risk to cover patterns that are too different of

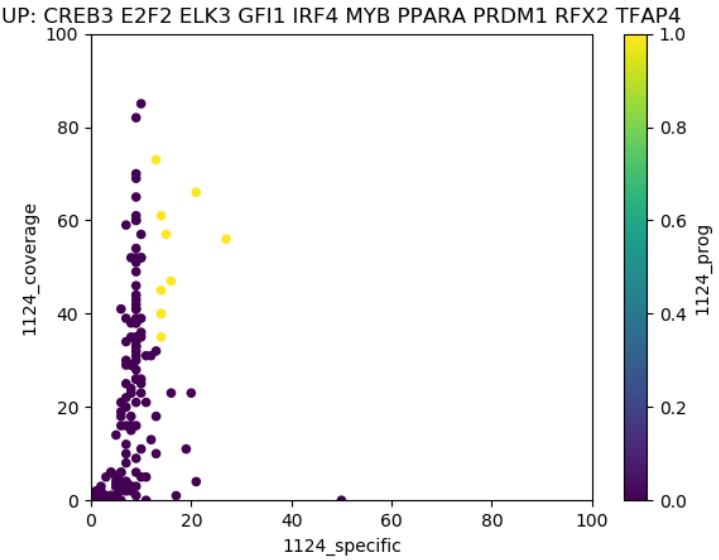
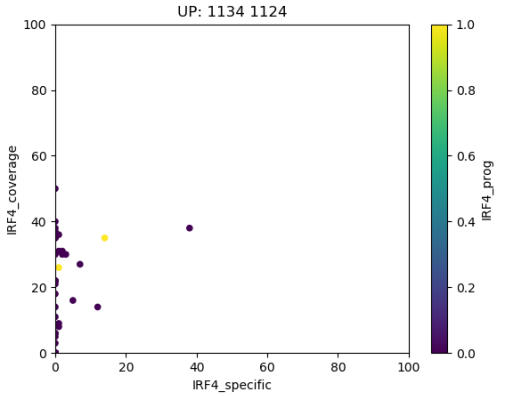


FIGURE 7.10 – Focusing on pattern: identification of putative regulators. Example on pattern 1124.

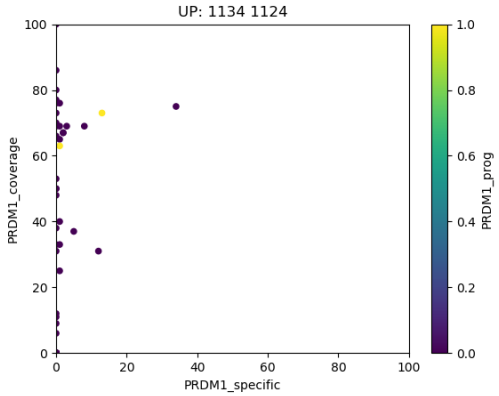
each other to find a global direction. But this representation is help-full for biologist as it allow to easily find the supposed target of a TF and the direction it influence.

7.4.4 Consistency with the literature

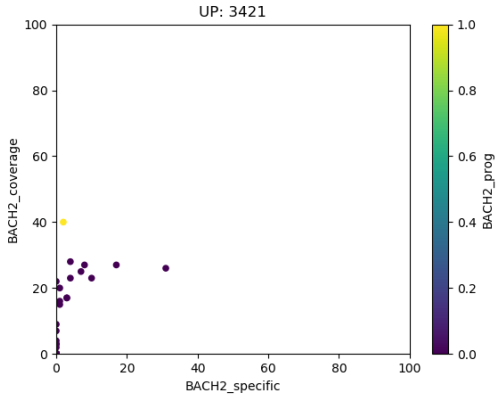
This method allow us to retrieve some known TF as TF of interest while they were not necessarily high in the coverage or specificity lists. The combination of both parameters allow us to filter some highly ubiquitous TF (for example: SP4 which has a high coverage but very low specificity). This method was validated by the 3 mains TF of our model: PRDM1, IRF4 and BACH2. PRDM1 and IRF4 both being up in 1134 and 1124 as activators, while the literature describe them as activators of the PB identity. BACH2 is found as an activator in 3421 and is given as an inhibitor of the PB (last population of the pattern). PAX5 is alsoo found as an activator of 3421.



(a) Example of IRF4: up in 1134 and 1124

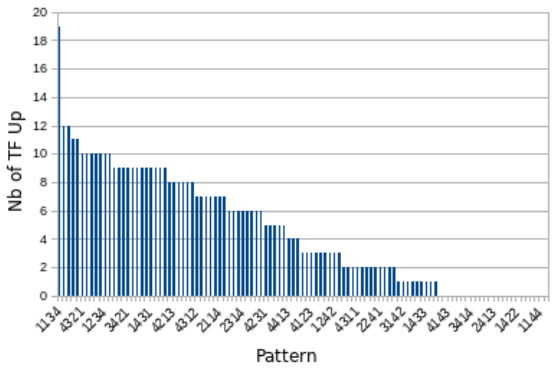


(b) Example of PRDM1: up in 1134 and 1124

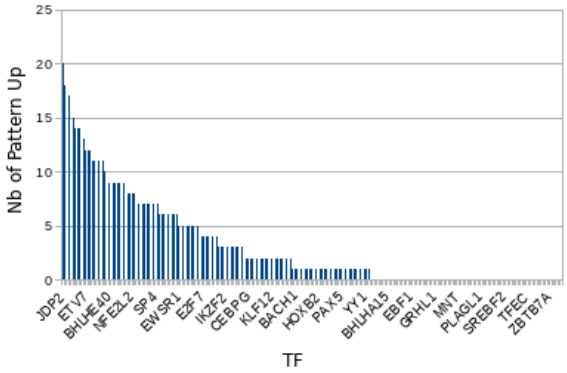


(c) Example of BACH2: up in 3421

FIGURE 7.11 – Focusing on TF: identification of expression patterns susceptible to be regulated by this TF.



(a) Number of TF passing both threshold by patterns.



(b) Number of patterns for which a TF pass both threshold.

FIGURE 7.12 – Number of elements passing the threshold by TF or by Pattern

7.5 Conclusion

As shown in this chapter we are able to infer signed networks on the B cells differentiation. We produce several regulatory networks, one by pattern and have also produced an overall graph of regulation focusing on the pattern level. We are also able to extract key regulators of different patterns and to reduce the search space for further biological experiments.

A first step in the validation of the obtained networks, was the occurrence of three TFs we were expecting to find according to the bibliography and with the same impact on the direction of the differentiation. But to further confirm those TF we need to see if they have a real impact on the regulatory network with hand-on experiments. Unfortunately, the biological experiments necessary to confirm the candidates TF are long and costly and we have not been able to perform them during the therm of this Ph.D.

CHAPITRE 8

CONCLUSION AND PERSPECTIVES

In this chapter we discuss the contributions presented in this thesis and the perspectives and future works related to it.

8.1 Contributions and limitations

In this thesis we focused on the regulatory network inference methods and their application to two specific biological contexts: the B cells differentiation and the follicular lymphoma.

Firstly, we revealed and analyzed the reproducibility and re-usability limitations of the most recent and complete network inference method: *Regulatory Circuits*. We tried to apply it to our data, since our cell populations do not appear in the published inferred networks. Following the methodology explained in the *Regulatory Circuits* article proved more complex than expected: some of the workflow steps were poorly described or plainly different between their explanation and the published data-sets they were supposed to generate. We proposed two ways of recomputing *Regulatory Circuits*, the first one using a maximum of the published files (intermediary and entry ones) and a second one using only the entry files. Both of these methods led to different results - varying both from one another and from the original networks. This unfortunately illustrates a common pitfall of reproducible and reusable science.

Secondly, to address the issue of re-usability, we proposed a new method for structuring and representing *Regulatory Circuits* using the Semantic Web technologies framework. We produced a RDF graph of the relations between the biological entities of *Regulatory Circuits*. We showed that the first layer of regulatory networks (sample-specific) provided by *Regulatory Circuits* can be generated by two SPARQL queries. At this point the method did not allow to easily extract the second layer of regulatory net-

works (tissue-specific) and needed post-computation to obtain them. We showed that Semantic Web technologies are a relevant framework to support network inference models and that they improved their re-usability.

To further improve the re-use of *Regulatory Circuits*, we thirdly proposed an extension of the previous method to generate the second layer of regulatory networks. Using the same RDF graph representing the biological data as described in the previous section, we could use the same two queries but this time we re-injected the result of the queries in the triplestore. The re-injected networks were put in specific named graphs - named after the sample they represent. We could then query those named graphs to generate the tissue-specific networks, and similarly re-inject them in another layer of named graphs. This allowed us to propose a resource composed of all the biological entities of *Regulatory Circuits*, but also of all the inferred regulatory networks, improving its re-usability. However, this was a very time-consuming task (a month for 808 sample-specific networks and 394 tissues). We are currently looking at other options to improve it efficiency such as not directly re-injecting the results of the queries but extracting them and using Quads [1] to inject the result in a second time.

As we saw that we could not reproduce *Regulatory Circuits* on new data, we fourthly proposed a new design for regulatory network inference. This method was designed to run on small, biologically-related data-sets and to produce signed networks. We took advantage of the heterogeneity of the biological data and used a multi-level normalisation of the activities based on the direction of the entities across all cell populations. We capitalized on the work previously done, as we developed a similar RDF graph-data reasoning. The design was based on expert knowledge to check the potential regulations and validate whether they are in concordance with the biological knowledge. We tested this method on data-set extracted from *Regulatory Circuits*, and obtained a better recall for the lowly expressed genes because we were able to take inhibitions into account. We also verified the signs of the relations with two major databases.

Fifthly, we applied this pipeline to four populations of B cells differentiation. We extracted 314,965 TF-genes relations representing two network levels: an overall network and several pattern-specific networks. To address the graphs high density, we proposed two criteria to select TF with a higher potential impact: the coverage (the ability of

1. https://www.w3.org/TR/n-quads/

the TF to have a large number of targets) and the specificity (the potential of the TF to regulate only a specific direction of the differentiation). This allowed us to identify a list of 146 TFs, including the major known TFs from the literature (BACH2, PRDM1, PAX5 and IRF4), and to associate each of them with the expression patterns they may regulate. This list still represents a large number of TF to further biologically experiment to validate their impact on the network.

8.2 Perspectives

In this section we present three main axes of perspectives: optimizing the methodology to find relevant TFs of significant impact on the network, better understanding the TFs mode of action at a bigger scale and not only in a specific relation, and the biological applications to follicular lymphoma, including the biological confirmation of the inferred TFs.

8.2.1 Improving the methodology to find significant TFs: finding minimum set of TF

The list of candidate TFs we propose is still composed of 146 TFs. We can narrow the search space by focusing on a specific pattern and its regulators, but this can still lead to up to 12 regulators. As biologically testing one TF is costly, we need to propose a way to further reduce this list of potential regulators.

In preliminary work, the solution approached was to find the minimum possible set of TF capable of covering all, or most part, of a given pattern. The goal was to give a small set of regulators to the biologists to test, and to have it be of the higher possible impact on the desired trajectory.

To do so, we chose to use ASP (Answer Set Programming). It took as input the list of TF included in the network, with the rule to use at least one and a set of rules representing the relations between TF and Genes. A rule was given as such: if TF is in the solution then exactly n out of the n genes its regulates are also in the solution. We could then ask to maximize the number of genes while minimizing the number of TF. We give an example of the type of ASP script used in FIGURE 8.1. This example

```
% Specify the TF set we can pick from:
1\{ tf(arhgef12); tf(arid3a); [...] ; tf(nfe2l1); tf(zkscan3); tf(atf6); tf(creb3l2)\}.

% Declaring the relations between TF and Genes:
15\{ gene(abhd17c); gene(aifm2);gene(atp8b2); gene(cenpu); gene(chek1); gene(dhrs9);
gene(gadd45a); gene(kcnn3); gene(mocs2); gene(mrpl46); gene(pmvk); gene(ppa1); gene(psma3);
gene(psmb1); gene(pus3) \}15:-tf(arhgef12).\\
150\{gene(abhd17c); gene(abhd2); gene(ada); [...] ; gene(yeats2); gene(zbed2) \}150:-tf(arid3a).

[...]

55{ gene(anxa6); gene(arhgap10); gene(arhgap18); gene(atox1); gene(atp8b2); [...] ; gene(yeats2) }55
:-tf(znf691).

Maximizing:
%{choose_TF(X):tf(X)}.
nb_TF(N):-N=\{ tf(\_)\}.
nb_Gene(Z):-Z= { gene(_)}.

#minimize{ S@1,S:nb_TF(S)}.
#maximize{ G@2,G:nb_Gene(G)}.
```

FIGURE 8.1 – Simplified ASP query to find solutions with the minimal number of TF and the maximal number of genes.

includes regulatory relations for the 1124 pattern at a previous development state of the pipeline, therefore it does not correspond to the reality of the current results.

In this configuration, we were able to run the ASP script and to find a solution composed of 3 TF covering all the 556 genes of the pattern. The solution was computed under 2 seconds as shown in TABLE 8.1. Out of the 3 TF found, 2 were part of the 5 TF with the most coverage and one was a ubiquitous TF not found among the 10 TF with the most coverage.

Models	1
Optimum	yes
Optimization	**-556 3**
Calls	1
Time	**1.123s**
Solving	0.61s
1st Model	0.01s
Unsat	0.00s
CPU Time	1.114s

TABLE 8.1 – Example of ASP output, for pattern 1124 (previous iteration) 3 TF for 100% of coverage: FOXP1 (4441), SP3 (4441) and CTCF (3124)

We still need to run this method on all the current networks to have a better landscape of the TFs we could submit to biological experiments. This approach allowed us to quickly find small sets of TF with large impact on specific networks, however it does not take into account if the output TF have a large variety of patterns' targets. It would be interesting to limit the search of the TFs with this methodology to the TFs previously found as interesting both in term of coverage and specificity (see SECTION 7.4).

8.2.2 Better understanding the TF roles in the overall regulatory network

In the current methodology developed during this thesis we mainly focus on the interactions between a TF and a gene or a TF on a pattern and eventually patterns onto patterns. We do not look at the overall graph, which could lead us to misinterpret some regulatory mechanisms on a higher level. With the overall network there is two aspects to take into consideration: the combinatory nature of the regulation and the consistency of the predicted network.

The combinatory aspect of the regulation means that there could be a cascade of events before the regulation we are looking at. For example: a TF we extracted to be interesting could itself be regulated by another TF. Therefore, the second TF, up-stream of the one we identified, could potentially have an higher impact on the

global network. But also than regulators do not necessarily act independently: there are events of competition between two TF, collaboration (example in [JOLMA et al., 2015]) or they can bind DNA sequentially.

To look at the cascade of regulation, we should look at the graph of regulations composed only by the 266 TF we consider and look at their interactions. We have only 7,723 interactions of a TF on another TF, where a complete graph would have been of 70,756 interactions, but as we can see in FIGURE 8.2 this still forms a highly connected graph. This is what we started to do in SECTION 7.3 with the interactions of TFs onto patterns.

For the second aspect (potential collaboration or competition between TFs), we will look to see if some TFs always appear together in the same patterns (either with the same or opposite regulatory input) in the resulting networks. We also need to further investigate the literature to try to find existing known collaboration or competition between TFs. We could also try to prove our hypothesis through experimentation.

In addition to the combinatory aspect, it is important to look at the overall consistency of the predicted networks. We know that the TF-genes relations are themselves consistent regarding the TF and the genes activities, but it does not mean they are consistent with other regulations surrounding them.

An example of consistency can be seen in [BAUMURATOVA et al., 2010]: if a TF is an activator of two genes and that only one is present in the population it is inconsistent. This could be explained by a difference of regulatory regions: one being closed and the other open in the same population, but it would be interesting to check. Another example would be activations' loop between two TFs but where one is not present in the given population.

To compute the overall consistency, we propose to use Caspo [VIDELA et al., 2017] to check the implemented rules of consistency.

8.2.3 Biological applications

The final goal of this work was to generate regulatory networks relevant for analyzing follicular lymphoma emergence. Our aim is to be able to decipher and prioritize non coding mutations in this context, by identifying their potential impact on regulatory networks. Therefore, we applied the pipeline to biological data relevant for follicular

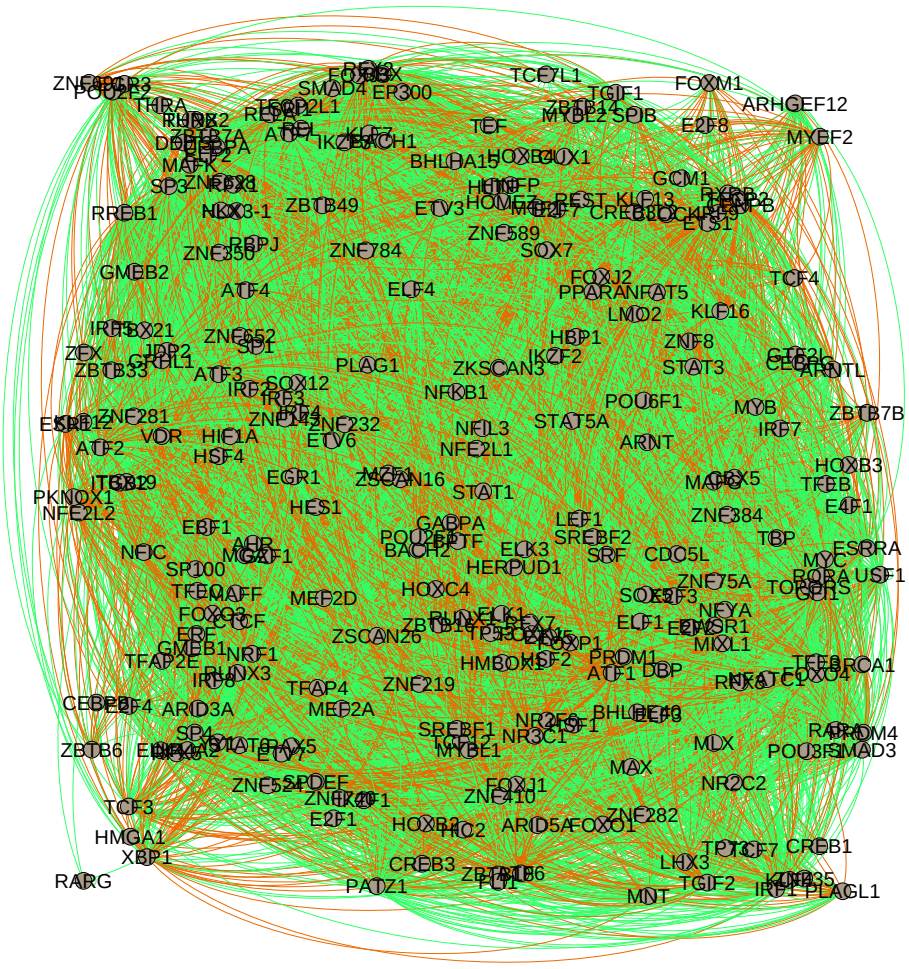


FIGURE 8.2 – Regulatory network, focusing only on TF/TF interactions. In green activation and in red inhibition.

lymphoma.

We selected five cell populations as inputs: Naive B cells (NBC), Centroblast (CB), Centrocyte (CC), Follicular Lymphoma (FL) and Plasmablast (PB). CB and CC are two germinal center B cell subsets, which are described as the origin of FL tumoral B cells. The data-set is composed of 26,802 genes, of which only 13,688 pass the differential expression filter and 83,240 regions obtained by ATAC-seq analysis. The rest of the computed and integrated elements are presented in TABLE 8.2.

gene_pattern	13,688
region_pattern	83,240
tf_pattern	350
gene_region_closest	1,192,688
tf_region_intersect	2,716,726

TABLE 8.2 – Numbers of entities integrated for the FL analysis

The pipeline produced 10,476,567 TF-region-gene relations before the compatibility table and 618,907 unique signed TF-gene relations after filtering.

While the regulatory networks have been produced, they are not yet analysed. A future work will be to comparatively analysed the networks obtained with the normal differentiation and with the follicular lymphoma. We hope to find changes in key regulators, by doing so we could then look into their binding sites to see if we can find mutations in them.

Finally, we are currently working with the INSERM team U1236 working on the B cells, to develop a protocol in order to biologically experiment on the TF we infer to confirm their involvement in the regulatory networks. One idea would be to perform whole-genome analysis of TF binding sites (ChIP-seq) in specific B cell subsets. A complementary approach could be to invalidate a potential regulator either by RNA interference or by genome editing (CRISPR / Cas9 technology), and to evaluate the consequences on B cell differentiation and/or on tumorigenic potential of primary B cells or cell lines.

Unfortunately, this is costly both in time and money, to order the antibody specific of the TF we are looking for. This would therefore necessitate to further reduce the list of provided TFs, for example using the ASP method proposed in a previous section.

APPENDIX

Transcription Factors passing the threshold in at least one Pattern

List of the TFs passing the threshold described in CHAPTER 7, SECTION 7.4.3 for at least one pattern. TFs are ordered in alphabetical order, and the ones in red are the ones known in the literature.

ARID3A, ARID5A, ATF1, BACH1, BACH2, BBX, BHLHE40, BHLHE41, CEBPB, CEBPG, CREB1, CREB3, CTCF, CUX1, DBP, E2F1, E2F2, E2F4, E2F7, EGR1, EGR2, EGR3, ELF1, ELF2, ELF4, ELK1, ELK3, ESR1, ESR2, ESRRA, ETS1, ETV6, ETV7, EWSR1, FLI1, FOXJ1, FOXJ2, FOXK1, FOXO1, FOXO3, FOXO4, GFI1, GTF2I, HBP1, HERPUD1, HINFP, HLX, HMGA1, HOXB2, HOXB3, HOXB4, HOXC4, HSF1, HSF2, HSF4, IKZF1, IKZF2, IRF1, IRF3, IRF4, IRF5, IRF7, IRF8, IRF9, JDP2, KLF12, KLF16, KLF4, KLF7, LEF1, LHX3, LMO2, MAFG, MAFK, MEF2A, MGA, MIXL1, MLX, MYB, MYBL1, MYEF2, NFATC1, NFE2L1, NFE2L2, NFIC, NFIL3, NFYA, NKX3-1, NR2F6, NR3C1, NR3C2, NR4A2, NRF1, PAX5, POU2F1, POU2F2, POU3F1, POU6F1, PPARA, PRDM1, PRDM4, RARA, RARG, REST, RFX2, RFX3, RFX5, RORA, RREB1, RUNX1, RUNX2, RXRA, SMAD3, SMAD4, SOX4, SOX5, SOX7, SP1, SP3, SP4, SPDEF, SPI1, SPIB, SRF, STAT3, STAT5A, TBX21, TCF12, TCF3, TCF7, TCF7L1, TFAP4, TFCP2, TFCP2L1, TGIF1, THRA, THRB, TP63, YY1, ZBTB14, ZEB1, ZFX, ZNF143, ZNF219, ZNF652, ZNF75A

Circuits	Nb of TF			Nb of Genes			Nb of Relations			% of complete graph		
	OG	Using Ranks	Re-computing	OG	Using Ranks	Re-computing	OG	Using Ranks	Re-computing	OG	Using Ranks	Re-computing
b_lymphoblastoid_cell_line	643	594	594	12330	11338	12305	1358031	602268	769386	17.1291429695807	8.94266353783023	10.5263114690177
brain_fetal	643	594	594	12218	11827	12065	407056	234251	239049	5.18135163503252	3.33442084097364	3.3355938163232
cd4+_t_cells	643	593	593	11911	10902	11881	1281420	526036	715736	16.7314007087036	8.13681788046997	10.158646148505
cd8+_t_cells	643	593	593	11919	10936	11893	1327365	555780	744147	17.3196682584115	8.57017557927096	10.551461606293
cd34+_stem_cells_-_adult_bone_marrow_derived	643	593	593	12151	11837	12085	733382	407965	416869	9.38657712125019	5.81201283710251	5.81698913192877
colon_adult	643	594	596	14850	12847	14812	1215042	301243	688446	12.7248849301726	3.94756050161982	7.79847919426839
colon_fetal	643	595	595	13890	13518	13787	675760	377168	384553	7.56622518410036	4.68927206923604	4.68780418528476
epitheloid_cancer_cell_line	643	595	595	12844	11145	11942	1256139	445225	619340	15.2099024881906	6.7140180432873	8.71635875921295
pancreas_adult	643	595	595	12703	12323	12567	476699	267037	271991	5.83615704596543	3.64198393793937	3.63752471626034
peripheral_blood_mononuclear_cells	643	592	593	12780	11473	12763	1796098	154354	1010008	21.8568817431981	2.27258004103642	13.3449622968163
small_intestine_adult	643	596	596	14710	14378	14677	1181926	653491	666581	12.4958740946003	7.62596612460685	7.62025275358926
small_intestine_fetal	643	595	595	14492	14119	14414	805147	445257	454427	8.64044043820605	5.30017063840906	5.29861840670776

TABLE 8.3 – Extension of TABLE 3.2, network by network. Comparison between the 3 ways of calculating *Regulatory Circuits*. Both method using or re-computing ranks produce networks that are included in the original *Regulatory Circuits* networks.

PUBLICATIONS

LOUARN, M., CHATONNET, F., GARNIER, X., FEST, T., SIEGEL, A. & DAMERON, O.(2019). *Increasing life science resources re-usability using Semantic Web technologies*, In Proceedings of the 15th IEEE International eScience conference,San Diego

BIBLIOGRAPHIE

ABUGESSAISA, I., SHIMOJI, H., SAHIN, S., KONDO, A., HARSHBARGER, J., LIZIO, M. Et al. (2016). FANTOM5 transcriptome catalog of cellular states based on Semantic MediaWiki. *Database*, *2016*.

AKASAKA, T., LOSSOS, I. S. & LEVY, R. (2003). BCL6 gene translocation in follicular lymphoma: a harbinger of eventual transformation to diffuse aggressive lymphoma. *Blood*, *102*(4), 1443-1448.

AL KAWAM, A., SEN, A., DATTA, A. & DICKEY, N. (2018). Understanding the Bioinformatics Challenges of Integrating Genomics into Healthcare. *IEEE journal of biomedical and health informatics*, *22*(5), 1672-1683. https://doi.org/https://doi.org/10.1109/JBHI.2017.2778263

ALDHOUS, P. (1993). Managing the genome data deluge. *Science (New York, N.Y.)*, *262*(5133), 502-503.

ANDERSSON, R., GEBHARD, C., MIGUEL-ESCALADA, I., HOOF, I., BORNHOLDT, J., BOYD, M. Et al. (2014). An atlas of active enhancers across human cell types and tissues. *Nature*, *507*(7493), 455.

ANDREU-PEREZ, J., POON, C. C., MERRIFIELD, R. D., WONG, S. T. & YANG, G.-Z. (2015). Big data for health. *IEEE journal of biomedical and health informatics*, *19*(4), 1193-1208.

ANTEZANA, E., KUIPER, M. & MIRONOV, V. (2009). Biological knowledge management: the emerging role of the Semantic Web technologies. *Briefings in bioinformatics*, *10*(4), 392-407.

BANSAL, M., GATTA, G. D. & DI BERNARDO, D. (2006). Inference of gene regulatory networks and compound mode of action from time course gene expression profiles. *Bioinformatics*, *22*(7), 815-822.

BARWICK, B. G. & AL. (2016). Plasma cell differentiation is coupled to division-dependent dna hypomethylation and gene regulation. *Nature Immunology*, *17(10)*, 1216-1225. https://doi.org/10.1038/ni.3519

BAUMURATOVA, T., SURDEZ, D., DELYON, B., STOLL, G., DELATTRE, O., RADULESCU, O. & SIEGEL, A. (2010). Localizing potentially active post-transcriptional regulations in the Ewing's sarcoma gene regulatory network. *BMC systems biology*, *4*(1), 146.

BERNERS-LEE, T. & HENDLER, J. (2001). Publishing on the semantic web. *Nature*, *410*(6832), 1023-1024.

BIZER, C., HEATH, T. & BERNERS LEE, T. (2009). Linked Data–The story so far. *International Journal on Semantic Web and Information Systems*, *5*(3), 1-22.

BLAKE, J. A. & BULT, C. J. (2006). Beyond the data deluge: Data integration and bio-ontologies. *Journal of Biomedical Informatics*, *39*(3), 314-320.

BOYLE, A. P., HONG, E. L., HARIHARAN, M., CHENG, Y., SCHAUB, M. A., KASOWSKI, M., KARCZEWSKI, K. J., PARK, J., HITZ, B. C., WENG, S. Et al. (2012). Annotation of functional variation in personal genomes using RegulomeDB. *Genome research*, *22*(9), 1790-1797.

BRANDIZI, M., SINGH, A., RAWLINGS, C. & HASSANI-PAK, K. (2018). Towards FAIRer Biological Knowledge Networks Using a Hybrid Linked Data and Graph Database Approach [In press]. *Journal of integrative bioinformatics*, *15*(3).

BUTTE, A. J. & KOHANE, I. S. (1999). Mutual information relevance networks: functional genomic clustering using pairwise entropy measurements, In *Biocomputing 2000*. World Scientific.

CANNATA, N., MERELLI, E. & ALTMAN, R. B. (2005). Time to Organize the Bioinformatics Resourceome. *PLoS Computational Biology*, *1*(7), 0531-0533.

CARBONE, A., ROULLAND, S., GLOGHINI, A., YOUNES, A., von KEUDELL, G., LÓPEZ-GUILLERMO, A. & FITZGIBBON, J. (2019). Follicular lymphoma. *Nature Reviews Disease Primers*, *5*(1), 1-20.

CARON, G. & AL. (2015). Cell-Cycle-Dependent Reconfiguration of the DNA Methylome during Terminal Differentiation of Human B Cells into Plasma Cells. *Cell Reports*, *13*, 1059-1071. https://doi.org/GSE72498

CHEN, H. & VANBUREN, V. (2012). A review of integration strategies to support gene regulatory network construction. *The Scientific World Journal*, *2012*, 435257.

CHEN, H., YU, T. & CHEN, J. Y. (2012). Semantic Web meets Integrative Biology: a survey. *Briefings in bioinformatics*, *14*(1), 109-125.

CONSORTIUM, E. P. Et al. (2012). An integrated encyclopedia of DNA elements in the human genome. *Nature*, *489*(7414), 57.

CORREIA, C., SCHNEIDER, P. A., DAI, H., DOGAN, A., MAURER, M. J., CHURCH, A. K., NOVAK, A. J., FELDMAN, A. L., WU, X., DING, H. Et al. (2015). BCL2 mutations are associated with increased risk of transformation and shortened survival in follicular lymphoma. *Blood, The Journal of the American Society of Hematology*, *125*(4), 658-667.

COTTLE, M., HOOVER, W., KANWAL, S., KOHN, M., STROME, T. & TREISTER, N. (2013). Transforming Health Care Through Big Data Strategies for leveraging big data in the health care industry. *Institute for Health Technology Transformation, http://ihealthtran.com/big-data-in-healthcare*.

DESMOTS, F., ROUSSEL, M., PANGAULT, C., LLAMAS-GUTIERREZ, F., PASTORET, C., GUIHENEUF, E., LE PRIOL, J., CAMARA-CLAYETTE, V., CARON, G., HENRY, C. Et al. (2019). Pan-HDAC inhibitors restore PRDM1 response to IL21 in CREBBP-mutated follicular lymphoma. *Clinical Cancer Research*, *25*(2), 735-746.

DOMINGUEZ, P. M., TEATER, M. & SHAKNOVICH, R. (2017). The new frontier of epigenetic heterogeneity in B-cell neoplasms. *Current Opinion in Hematology*, *24*(4), 402-408.

GARDNER, T. S., DI BERNARDO, D., LORENZ, D. & COLLINS, J. J. (2003). Inferring genetic networks and identifying compound mode of action via expression profiling. *Science*, *301*(5629), 102-105.

GARNIS, C., BUYS, T. P. & LAM, W. L. (2004). Genetic alteration and gene expression modulation during cancer progression. *Molecular Cancer*, *3*(1), 9.

GERSTEIN, M. B., KUNDAJE, A., HARIHARAN, M., LANDT, S. G., YAN, K.-K., CHENG, C. Et al. (2012). Architecture of the human regulatory network derived from ENCODE data. *Nature*, *489*(7414), 91.

GOBLE, C. & STEVENS, R. (2008). State of the nation in data integration for bioinformatics. *Journal of biomedical informatics*, *41*(5), 687-693.

HAN, H., CHO, J.-W., LEE, S., YUN, A., KIM, H., BAE, D., YANG, S., KIM, C. Y., LEE, M., KIM, E. Et al. (2018). TRRUST v2: an expanded reference database of human and mouse transcriptional regulatory interactions. *Nucleic acids research*, *46*(D1), D380-D386.

HARTEMINK, A. J., GIFFORD, D. K., JAAKKOLA, T. S. & YOUNG, R. A. (2000). Using graphical models and genomic expression data to statistically validate models of genetic regulatory networks, In *Biocomputing 2001*. World Scientific.

HASIN, Y., SELDIN, M. & LUSIS, A. (2017). Multi-omics approaches to disease. *Genome biology*, *18*(1), 1-15.

HAURY, A.-C., MORDELET, F., VERA-LICONA, P. & VERT, J.-P. (2012). TIGRESS: trustful inference of gene regulation using stability selection. *BMC systems biology*, *6*(1), 145.

HECKER, M., LAMBECK, S., TOEPFER, S., VAN SOMEREN, E. & GUTHKE, R. (2009). Gene regulatory network inference: data integration in dynamic models—a review. *Biosystems*, *96*(1), 86-103.

HIPP, N., SYMINGTON, H., PASTORET, C., CARON, G., MONVOISIN, C., TARTE, K., FEST, T. & DELALOY, C. (2017). IL-2 imprints human naive B cell fate towards plasma cell through ERK/ELK1-mediated BACH2 repression. *Nature communications*, *8*(1), 1-17.

HUTTENHOWER, C., MUTUNGU, K. T., INDIK, N., YANG, W., SCHROEDER, M., FORMAN, J. J., TROYANSKAYA, O. G. & COLLER, H. A. (2009). Detailing regulatory networks through large scale data integration. *Bioinformatics*, *25*(24), 3267-3274.

JIANG, Y., DOMINGUEZ, P. M. & MELNICK, A. M. (2016). The many layers of epigenetic dysfunction in B-cell lymphomas. *Current opinion in hematology*, *23*(4), 377-384.

JOLMA, A., YIN, Y., NITTA, K. R., DAVE, K., POPOV, A., TAIPALE, M., ENGE, M., KIVIOJA, T., MORGUNOVA, E. & TAIPALE, J. (2015). DNA-dependent formation of transcription factor pairs alters their binding specificity. *Nature*, *527*(7578), 384-388.

JUPP, S., MALONE, J., BOLLEMAN, J., BRANDIZI, M., DAVIES, M., GARCIA, L. Et al. (2014). The EBI RDF platform: linked open data for the life sciences. *Bioinformatics*, *30*(9), 1338-1339.

KALDERIMIS, A., LYNE, R., BUTANO, D., CONTRINO, S., LYNE, M., HEIMBACH, J. Et al. (2014). InterMine: extensive web services for modern biology. *Nucleic Acids Res.*, *42*(Web Server issue), W468-472.

KAMDAR, M. R., FERNÁNDEZ, J. D., POLLERES, A., TUDORACHE, T. & MUSEN, M. A. (2019). Enabling Web-scale data integration in biomedicine through Linked Open Data. *NPJ digital medicine*, *2*, 90. https://doi.org/https://doi.org/10.1038/s41746-019-0162-5

KARCZEWSKI, K. J. & SNYDER, M. P. (2018). Integrative omics for health and disease. *Nature Reviews Genetics*, *19*(5), 299.

KAROLCHIK, D., BARBER, G. P., CASPER, J., CLAWSON, H., CLINE, M. S., DIEKHANS, M., DRESZER, T. R., FUJITA, P. A., GURUVADOO, L., HAEUSSLER, M. Et al. (2014). The UCSC genome browser database: 2014 update. *Nucleic acids research*, *42*(D1), D764-D770.

KENT, W. J., SUGNET, C. W., FUREY, T. S., ROSKIN, K. M., PRINGLE, T. H., ZAHLER, A. M. & HAUSSLER, D. (2002). The human genome browser at UCSC. *Genome research*, *12*(6), 996-1006.

KHERADPOUR, P., ERNST, J., MELNIKOV, A., ROGOV, P., WANG, L., ZHANG, X., ALSTON, J., MIKKELSEN, T. S. & KELLIS, M. (2013). Systematic dissection of regulatory motifs in 2000 predicted human enhancers using a massively parallel reporter assay. *Genome research*, *23*(5), 800-811.

KHERADPOUR, P. & KELLIS, M. (2014). Systematic discovery and characterization of regulatory motifs in ENCODE TF binding experiments. *Nucleic acids research*, *42*(5), 2976-2987.

KHERADPOUR, P., STARK, A., ROY, S. & KELLIS, M. (2007). Reliable prediction of regulator targets using 12 Drosophila genomes. *Genome research*, *17*(12), 1919-1931.

KHURANA, E., FU, Y., CHAKRAVARTY, D., DEMICHELIS, F., RUBIN, M. A. & GERSTEIN, M. (2016). Role of non-coding sequence variants in cancer. *Nature Reviews Genetics*, *17*(2), 93.

KORFI, K., ALI, S., HEWARD, J. A. & FITZGIBBON, J. (2017). Follicular lymphoma, a B cell malignancy addicted to epigenetic mutations. *Epigenetics*, *12*(5), 370-377.

KUNDAJE, A., MEULEMAN, W., ERNST, J., BILENKY, M., YEN, A., HERAVI-MOUSSAVI, A. Et al. (2015). Integrative analysis of 111 reference human epigenomes. *Nature*, *518*(7539), 317.

LANGFELDER, P. & HORVATH, S. (2008). WGCNA: an R package for weighted correlation network analysis. *BMC bioinformatics*, *9*(1), 559.

LAWRENCE, C. E. & REILLY, A. A. (1990). An expectation maximization (EM) algorithm for the identification and characterization of common sites in unaligned biopolymer sequences. *Proteins: Structure, Function, and Bioinformatics*, *7*(1), 41-51.

LEMMENS, K., DE BIE, T., DHOLLANDER, T., DE KEERSMAECKER, S. C., THIJS, I. M., SCHOOFS, G., DE WEERDT, A., DE MOOR, B., VANDERLEYDEN, J., COLLADO-VIDES, J. Et al. (2009). DISTILLER: a data integration framework to reveal

condition dependency of complex regulons in Escherichia coli. *Genome biology*, *10*(3), R27.

LIANG, S., FUHRMAN, S., SOMOGYI, R. Et al. (1998). Reveal, a general reverse engineering algorithm for inference of genetic network architectures, In *Pacific symposium on biocomputing*.

LICATA, L., LO SURDO, P., IANNUCCELLI, M., PALMA, A., MICARELLI, E., PERFETTO, L., PELUSO, D., CALDERONE, A., CASTAGNOLI, L. & CESARENI, G. (2020). SIGNOR 2.0, the SIGnaling Network Open Resource 2.0: 2019 update. *Nucleic acids research*, *48*(D1), D504-D510.

LIVINGSTON, K. M., BADA, M., HUNTER, L. E. & VERSPOOR, K. (2013). Representing annotation compositionality and provenance for the Semantic Web. *Journal of biomedical semantics*, *4*, 38.

LIZIO, M., HARSHBARGER, J., SHIMOJI, H., SEVERIN, J., KASUKAWA, T., SAHIN, S. Et al. (2015). Gateways to the FANTOM5 promoter level mammalian expression atlas. *Genome biology*, *16*(1), 22.

LONSDALE, J., THOMAS, J., SALVATORE, M., PHILLIPS, R., LO, E., SHAD, S., HASZ, R., WALTERS, G., GARCIA, F., YOUNG, N. Et al. (2013). The genotype-tissue expression (GTEx) project. *Nature genetics*, *45*(6), 580.

LOUARN, M., CHATONNET, F., GARNIER, X., FEST, T., SIEGEL, A. & DAMERON, O. (2019). Increasing life science resources re-usability using Semantic Web technologies, In *Proceedings of the 15th IEEE International eScience conference, San Diego*.

MADAN BABU, M. & TEICHMANN, S. A. (2003). Evolution of transcription factors and the gene regulatory network in Escherichia coli. *Nucleic acids research*, *31*(4), 1234-1244.

MADAR, A., GREENFIELD, A., VANDEN-EIJNDEN, E. & BONNEAU, R. (2010). DREAM3: network inference using dynamic context likelihood of relatedness and the inferelator. *PloS one*, *5*(3), e9803.

MALLADI, V. S., ERICKSON, D. T., PODDUTURI, N. R., ROWE, L. D., CHAN, E. T., DAVIDSON, J. M. Et al. (2015). Ontology application and use at the ENCODE DCC. *Database*, *2015*.

MANSOUR, M. R., ABRAHAM, B. J., ANDERS, L., BEREZOVSKAYA, A., GUTIERREZ, A., DURBIN, A. D., ETCHIN, J., LAWTON, L., SALLAN, S. E., SILVERMAN, L. B. Et

al. (2014). An oncogenic super-enhancer formed through somatic mutation of a noncoding intergenic element. *Science*, *346*(6215), 1373-1377.

MARBACH, D., LAMPARTER, D., QUON, G., KELLIS, M., KUTALIK, Z. & BERGMANN, S. (2016). Tissue-specific regulatory circuits reveal variable modular perturbations across complex diseases. *Nature methods*, *13*(4), 366.

MARGOLIN, A. A., NEMENMAN, I., BASSO, K., WIGGINS, C., STOLOVITZKY, G., DALLA FAVERA, R. & CALIFANO, A. (2006). ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context, In *BMC bioinformatics*. Springer.

MAUMET, C., AUER, T., BOWRING, A., CHEN, G., DAS, S., FLANDIN, G. Et al. (2016). Sharing brain mapping statistical results with the neuroimaging data model. *Scientific data*, *3*, 160102.

MEYER, L. R., ZWEIG, A. S., HINRICHS, A. S., KAROLCHIK, D., KUHN, R. M., WONG, M., SLOAN, C. A., ROSENBLOOM, K. R., ROE, G., RHEAD, B. Et al. (2013). The UCSC Genome Browser database: extensions and updates 2013. *Nucleic acids research*, *41*(D1), D64-D69.

NARLIKAR, G. J., FAN, H.-Y. & KINGSTON, R. E. (2002). Cooperation between complexes that regulate chromatin structure and transcription. *Cell*, *108*(4), 475-487.

ORICCHIO, E., CIRIELLO, G., JIANG, M., BOICE, M. H., SCHATZ, J. H., HEGUY, A., VIALE, A., de STANCHINA, E., TERUYA-FELDSTEIN, J., BOUSKA, A. Et al. (2014). Frequent disruption of the RB pathway in indolent follicular lymphoma suggests a new combination therapy. *Journal of Experimental Medicine*, *211*(7), 1379-1391.

PAPILI GAO, N., UD-DEAN, S. M., GANDRILLON, O. & GUNAWAN, R. (2018). SINCERITIES: inferring gene regulatory networks from time-stamped single cell transcriptional expression profiles. *Bioinformatics*, *34*(2), 258-266.

PÉREZ, J., ARENAS, M. & GUTIERREZ, C. (2009). Semantics and complexity of SPARQL. *ACM Transactions on Database Systems (TODS)*, *34*(3), 1-45.

PETRALIA, F., WANG, P., YANG, J. & TU, Z. (2015). Integrative random forest for gene regulatory network inference. *Bioinformatics*, *31*(12), i197-i205.

PHAN, T. G. & TANGYE, S. G. (2017). Memory B cells: total recall. *Current Opinion in Immunology*, *45*, 132-140.

QUEIRÓS, A. C., BEEKMAN, R., VILARRASA-BLASI, R., DURAN-FERRER, M., CLOT, G., MERKEL, A., RAINERI, E., RUSSIÑOL, N., CASTELLANO, G., BEÀ, S. Et al.

(2016). Decoding the DNA methylome of mantle cell lymphoma in the light of the entire B cell lineage. *Cancer cell*, *30*(5), 806-821.

QUINLAN, A. R. (2014). BEDTools: the Swiss-army tool for genome feature analysis. *Current protocols in bioinformatics*, 11-12.

REN, B., ROBERT, F., WYRICK, J. J., APARICIO, O., JENNINGS, E. G., SIMON, I., ZEITLINGER, J., SCHREIBER, J., HANNETT, N., KANIN, E. Et al. (2000). Genome-wide location and function of DNA binding proteins. *Science*, *290*(5500), 2306-2309.

RIGDEN, D. J. & FERNÁNDEZ, X. M. (2019). The 26th annual Nucleic Acids Research database issue and Molecular Biology Database Collection. *Nucleic acids research*, *47*(D1), D1-D7. https://doi.org/https://doi.org/10.1093/nar/gky1267

RODRÍGUEZ-IGLESIAS, A., RODRÍGUEZ-GONZÁLEZ, A., IRVINE, A. G., SESMA, A., URBAN, M., HAMMOND-KOSACK, K. E. & WILKINSON, M. D. (2016). Publishing FAIR Data: An Exemplar Methodology Utilizing PHI-Base. *Frontiers in plant science*, *7*, 641.

ROY, G. G., GEARD, N., VERSPOOR, K. & HE, S. (2020). PoLoBag: Polynomial Lasso Bagging for signed gene regulatory network inference from expression data. *Bioinformatics*.

SEIFERT, M. & AL. (2015). Functional capacities of human igm memory b cells in early inflammatory responses and secondary germinal center reactions. *PNAS*, *112*(6), 546-555. https://doi.org/10.1073/pnas.1416276112

SINHA, S. & TOMPA, M. (2000). A statistical method for finding transcription factor binding sites., In *ISMB*.

SINHA, S. & TOMPA, M. (2003). YMF: a program for discovery of novel transcription factor binding sites by statistical overrepresentation. *Nucleic acids research*, *31*(13), 3586-3588.

SKIPPER, M., ECCLESTON, A., GRAY, N., HEEMELS, T., LE BOT, N., MARTE, B. & WEISS, U. (2015). Presenting the Epigenome Roadmap. *Nature*, *518*, 313.

SMALLWOOD, A. & REN, B. (2013). Genome organization and long-range regulation of gene expression by enhancers. *Current opinion in cell biology*, *25*(3), 387-394.

SMEDLEY, D., HAIDER, S., DURINCK, S., PANDINI, L., PROVERO, P., ALLEN, J. Et al. (2015). The BioMart community portal: an innovative alternative to large, centralized data repositories. *Nucleic Acids Res.*, *43*(W1), W589-598.

SNYDER, M. P., GINGERAS, T. R., MOORE, J. E., WENG, Z., GERSTEIN, M. B., REN, B., HARDISON, R. C., STAMATOYANNOPOULOS, J. A., GRAVELEY, B. R., FEINGOLD, E. A. Et al. (2020). Perspectives on ENCODE. *Nature*, *583*(7818), 693-698.

STEIN, L. D. (2003). Integrating biological databases. *Nature reviews. Genetics*, *4*(5), 337-345.

STEPHENS, S., LAVIGNA, D., DILASCIO, M. & LUCIANO, J. (2006). Aggregation of Bioinformatics Data Using Semantic Web Technology. *Journal of Web Semantics*, *4*(3).

STEPHENS, Z. D., LEE, S. Y., FAGHRI, F., CAMPBELL, R. H., ZHAI, C., EFRON, M. J., IYER, R., SCHATZ, M. C., SINHA, S. & ROBINSON, G. E. (2015). Big Data: Astronomical or Genomical? *PLoS biology*, *13*(7), e1002195.

THOMPSON, W., ROUCHKA, E. C. & LAWRENCE, C. E. (2003). Gibbs Recursive Sampler: finding transcription factor binding sites. *Nucleic acids research*, *31*(13), 3580-3585.

VIDELA, S., SAEZ-RODRIGUEZ, J., GUZIOLOWSKI, C. & SIEGEL, A. (2017). caspo: a toolbox for automated reasoning on the response of logical signaling networks families. *Bioinformatics*, *33*(6), 947-950.

WHETZEL, P. L., NOY, N. F., SHAH, N. H., ALEXANDER, P. R., NYULAS, C., TUDORACHE, T. & MUSEN, M. A. (2011). BioPortal: enhanced functionality via new Web services from the National Center for Biomedical Ontology to access and use ontologies in software applications. *Nucleic acids research*, *39*(Web Server issue), W541-W545.

WILKINSON, M. D., DUMONTIER, M., AALBERSBERG, I. J. J., APPLETON, G., AXTON, M., BAAK, A. Et al. (2016). The FAIR Guiding Principles for scientific data management and stewardship. *Scientific data*, *3*, 160018.

WILLIS, S. N. & NUTT, S. L. (2019). New players in the gene regulatory network controlling late B cell differentiation. *Current Opinion in Immunology*, *58*, 68-74.

YILDIZ, M., LI, H., BERNARD, D., AMIN, N. A., OUILLETTE, P., JONES, S., SAIYA-CORK, K., PARKIN, B., JACOBI, K., SHEDDEN, K. Et al. (2015). Activating STAT6 mutations in follicular lymphoma. *Blood, The Journal of the American Society of Hematology*, *125*(4), 668-679.

ZAHN-ZABAL, M., MICHEL, P.-A., GATEAU, A., NIKITIN, F., SCHAEFFER, M., AUDOT, E., GAUDET, P., DUEK, P. D., TEIXEIRA, D., RECH DE LAVAL, V. Et al. (2020).

The neXtProt knowledgebase in 2020: data, tools and usability improvements. *Nucleic Acids Research*, *48*(D1), D328-D334.

UNIVERSITÉ DE RENNES 1

Titre: **Analyse et intégration de données génomiques larges et hétérogènes**

Mot clés : Bio-informatique, technologies du web sémantique, inférence de réseaux de régulations

Résumé: L'inférence de réseaux de régulation à partir de données hétérogènes a pour but d'identifier les régulateurs clefs impliqués dans des processus biologiques aboutissant à des cancers. Dans cette thèse, je m'intéresse à la différenciation des cellules B naïves, d'où émerge le lymphome folliculaire. Ma première contribution souligne les problèmes de réutilisation et de reproductibilité des méthodes d'inférence de réseaux actuelles. Pour surmonter ces limites, je propose une structure utilisant les technologies du Web Sémantique pour intégrer et requêter ces jeux de données hétérogènes de manière systématique (deuxième contribution). Le pipeline d'origine est reproduit par des requêtes sur le graphe de données, ce résultat peut lui-même être intégré et enrichi avec des données publiques (troisième contribution). Ceci démontre l'utilité de cette approche et de ses bénéfices en terme de réutilisation et de reproductibilité. Ma quatrième contribution est une nouvelle méthode d'inférence de réseaux prenant en compte la connaissance des experts, pour étendre l'analyse à des jeux de données restreints et biologiquement proches et pour introduire la notion de relations signées, incluant les inhibitions. Enfin, l'application de cette méthode à la différenciation des cellules B, a permis la découverte de 146 FT avec un impact potentiel majeur sur le réseau (cinquième contribution).

Title: **Analysis and integration of heterogeneous large-scale genomics data**

Keywords : BioInformatics, Semantic Web technologies, regulatory network inference

Abstract: Regulatory networks inference from heterogeneous data is a computational step aiming at identifying key regulators involved in differentiation processes leading to cancer. In this thesis I focus on B cell differentiation, from which follicular lymphoma emerges. The first contribution outlines the reproducibility and reusability limitations of a state-of-the-art method for network inference from genomic data. To overcome these limitations, I demonstrated that Semantic Web technologies can structure and integrate large-scale heterogeneous datasets in a systematic way (second contribution). The original analysis workflow outputs could be reproduced as queries on a graph of data, which could itself be layered and enriched with public databases (third contribution). This demonstrates the technical relevance of this approach and underlines its benefits in improving reusability and reproducibility. As a fourth contribution, a new method for network inference was designed to take expert knowledge into account - both to extend the previous framework to the analysis of smaller, closely-related datasets and to enrich the inferred networks with signs, therefore including inhibitory regulatory processes. Finally, the method was applied to B cell differentiation, leading to the discovery of 146 TF with potential large impact on the network (fifth contribution).