

UNIVERSITÉ PARIS DESCARTES

ED 474 Frontières du vivant

*Institut Curie, PSL Research University, Mines Paris Tech, Inserm U900
Centre de Recherches Interdisciplinaires
Paris, France*

**Unsupervised deconvolution of bulk omics
profiles: methodology and application to
characterize the immune landscape in tumors**
par Urszula Czerwińska

Thèse de doctorat Interdisciplinaire

Thèse dirigée par Andrei Zinovyev et Vassili Soumelis

Présentée et soutenue publiquement le 2 octobre 2018

Devant un jury composé de :

Andrei ZINOVYEV	directeur de thèse - Paris 5 Descartes
Vassili SOUMELIS	directeur de thèse - Paris 7 Diderot
Christophe AMBROISE	rapporteur - Université d'Evry Val d'Essonne
Aurélien DE REYNIÈS	rapporteur - Université Paris 6 Pierre et Marie Curie
Jean-Yves BLAY	examinateur - Université Lyon 1
Marielle CHIRON	examinatrice - Sanofi
Marie-Caroline DIEU-NOSJEAN	examinatrice - Université Paris 6 Pierre et Marie Curie
Daniel GAUTHERET	examinateur - Université Paris Sud



Except where otherwise noted, this work is licensed under
<http://creativecommons.org/licenses/by-nc-nd/3.0/>

Title: Déconvolution non supervisée des profils omiques de masse: méthodologie et application à la caractérisation du paysage immunitaire des tumeurs

Résumé (français) :

Les tumeurs sont entourées d'un microenvironnement complexe comprenant des cellules tumorales, des fibroblastes et une diversité de cellules immunitaires. Avec le développement actuel des immunothérapies, la compréhension de la composition du microenvironnement tumoral est d'une importance critique pour effectuer un pronostic sur la progression tumorale et sa réponse au traitement. Cependant, nous manquons d'approches quantitatives fiables et validées pour caractériser le microenvironnement tumoral, facilitant ainsi le choix de la meilleure thérapie.

Une partie de ce défi consiste à quantifier la composition cellulaire d'un échantillon tumoral (appelé problème de déconvolution dans ce contexte), en utilisant son profil omique de masse (le profil quantitatif global de certains types de molécules, tels que l'ARNm ou les marqueurs épigénétiques). La plupart des méthodes existantes utilisent des signatures prédéfinies de types cellulaires et ensuite extrapolent cette information à des nouveaux contextes. Cela peut introduire un biais dans la quantification de microenvironnement tumoral dans les situations où le contexte étudié est significativement différent de la référence.

Sous certaines conditions, il est possible de séparer des mélanges de signaux complexes, en utilisant des méthodes de séparation de sources et de réduction des dimensions, sans définitions de sources préexistantes. Si une telle approche (déconvolution non supervisée) peut être appliquée à des profils omiques de masse de tumeurs, cela permettrait d'éviter les biais contextuels mentionnés précédemment et fournirait un aperçu des signatures cellulaires spécifiques au contexte.

Dans ce travail, j'ai développé une nouvelle méthode appelée DeconICA (Déconvolution de données omiques de masse par l'analyse en composantes immunitaires), basée sur la méthodologie de séparation aveugle de source. DeconICA a pour but l'interprétation et la quantification des signaux biologiques, façonnant les profils omiques d'échantillons tumoraux ou de tissus normaux, en mettant l'accent sur les signaux liés au système immunitaire et la découverte de nouvelles signatures.

Afin de rendre mon travail plus accessible, j'ai implémenté la méthode DeconICA en tant que librairie R. En appliquant ce logiciel aux jeux de données de référence, j'ai démontré qu'il est possible de quantifier les cellules immunitaires avec une précision comparable aux méthodes de pointe publiées, sans définir a priori des gènes spécifiques au type cellulaire. DeconICA peut fonctionner avec des techniques de factorisation matricielle telles que l'analyse indépendante des composants (ICA) ou la factorisation matricielle

non négative (NMF).

Enfin, j'ai appliqué DeconICA à un grand volume de données : plus de 100 jeux de données, contenant au total plus de 28 000 échantillons de 40 types de tumeurs, générés par différentes technologies et traités indépendamment. Cette analyse a démontré que les signaux immunitaires basés sur l'ICA sont reproductibles entre les différents jeux de données. D'autre part, nous avons montré que les trois principaux types de cellules immunitaires, à savoir les lymphocytes T, les lymphocytes B et les cellules myéloïdes, peuvent y être identifiés et quantifiés.

Enfin, les métagènes dérivés de l'ICA, c'est-à-dire les valeurs de projection associées à une source, ont été utilisés comme des signatures spécifiques permettant d'étudier les caractéristiques des cellules immunitaires dans différents types de tumeurs. L'analyse a révélé une grande diversité de phénotypes cellulaires identifiés ainsi que la plasticité des cellules immunitaires, qu'elle soit dépendante ou indépendante du type de tumeur. Ces résultats pourraient être utilisés pour identifier des cibles médicamenteuses ou des biomarqueurs pour l'immunothérapie du cancer.

Title: Unsupervised deconvolution of bulk omics profiles: methodology and application to characterize the immune landscape in tumors

Abstract: Tumors are engulfed in a complex microenvironment (TME) including tumor cells, fibroblasts, and a diversity of immune cells. Currently, a new generation of cancer therapies based on modulation of the immune system response is in active clinical development with first promising results. Therefore, understanding the composition of TME in each tumor case is critically important to make a prognosis on the tumor progression and its response to treatment. However, we lack reliable and validated quantitative approaches to characterize the TME in order to facilitate the choice of the best existing therapy.

One part of this challenge is to be able to quantify the cellular composition of a tumor sample (called deconvolution problem in this context), using its bulk omics profile (global quantitative profiling of certain types of molecules, such as mRNA or epigenetic markers). In recent years, there was a remarkable explosion in the number of methods approaching this problem in several different ways. Most of them use pre-defined molecular signatures of specific cell types and extrapolate this information to previously unseen contexts. This can bias the TME quantification in those situations where the context under study is significantly different from the reference.

In theory, under certain assumptions, it is possible to separate complex signal mixtures,

using classical and advanced methods of source separation and dimension reduction, without pre-existing source definitions. If such an approach (unsupervised deconvolution) is feasible to apply for bulk omic profiles of tumor samples, then this would make it possible to avoid the above mentioned contextual biases and provide insights into the context-specific signatures of cell types.

In this work, I developed a new method called DeconICA (Deconvolution of bulk omics datasets through Immune Component Analysis), based on the blind source separation methodology. DeconICA has an aim to decipher and quantify the biological signals shaping omics profiles of tumor samples or normal tissues. A particular focus of my study was on the immune system-related signals and discovering new signatures of immune cell types.

In order to make my work more accessible, I implemented the DeconICA method as an R package named “DeconICA”. By applying this software to the standard benchmark datasets, I demonstrated that DeconICA is able to quantify immune cells with accuracy comparable to published state-of-the-art methods but without a priori defining a cell type-specific signature genes. The implementation can work with existing deconvolution methods based on matrix factorization techniques such as Independent Component Analysis (ICA) or Non-Negative Matrix Factorization (NMF).

Finally, I applied DeconICA to a big corpus of data containing more than 100 transcriptomic datasets composed of, in total, over 28000 samples of 40 tumor types generated by different technologies and processed independently. This analysis demonstrated that ICA-based immune signals are reproducible between datasets and three major immune cell types: T-cells, B-cells and Myeloid cells can be reliably identified and quantified.

Additionally, I used the ICA-derived metagenes as context-specific signatures in order to study the characteristics of immune cells in different tumor types. The analysis revealed a large diversity and plasticity of immune cells dependent and independent on tumor type. Some conclusions of the study can be helpful in identification of new drug targets or biomarkers for immunotherapy of cancer.

Mots-clés (français) : microenvironnement tumoral, biologie des systèmes de cancer, analyse de données omiques, analyse de données monocellulaires, bioinformatique, hétérogénéité, séparation aveugle de source, apprentissage non supervisé, cancer, oncologie, immunologie

Keywords: tumor microenvironment, cancer systems biology, omic data analysis, single cell data analysis, bioinformatics, heterogeneity, blind sources separation, unsupervised learning, cancer, oncology, immunology

Dédicace

À Richard

Avertissement

Cette thèse de doctorat est le fruit d'un travail approuvé par le jury de soutenance et réalisé dans le but d'obtenir le diplôme d'Etat de docteur de philosophie. Ce document est mis à disposition de l'ensemble de la communauté universitaire élargie. Il est soumis à la propriété intellectuelle de l'auteur. Ceci implique une obligation de citation et de référencement lors de l'utilisation de ce document. D'autre part, toute contrefaçon, plagiat, reproduction illicite encourt toute poursuite pénale.

Code de la Propriété Intellectuelle. Articles L 122.4

Code de la Propriété Intellectuelle. Articles L 335.2-L 335.10

Remerciments

I would like to thank my supervisors Andrei Zinovyev and Vassili Soumelis for guiding this project and enabling me to interact with their teams and sharing the resources. I would also like to thank the U900 lab and his head Emmanuel Barillot to generously equip me with the professional environment, the place and the tools.

I address my gratitude to the TAC committee members Franck Pagès and Denis Thieffry for helping me organizing the jury and giving constructive comments along with my thesis, for being present, at least remotely despite severe weather conditions or travels.

I would also like to express gratitude towards the jury for taking the time to assess this work.

This is also the place to thank the ITMO Cancer - AVIESAN for funding my Ph.D. scholarship and the Pharmacology Faculty of Paris Descartes and specifically Prof. Chantal Guihenneuc for giving me the opportunity to teach in parallel of my Ph.D. as a part of her pedagogical team. I would also thank a lot Center of Interdisciplinary Research for equipping me with unusual skills through numerous courses and Bettencourt Foundation for financing part of the training and sponsoring travel expenses. Special thanks to FdV coordinators: Sofie Leon, David Manset, Elodie Kaslikowski and Maria Molina Calvita for their availability and dynamism. Also, I would express my gratitude for supporting my application for French nationality to François Taddei, director of the FdV Ph.D. school.

Thanks to all people I worked with in both teams: Arnau, Pauline, Gaelle, Paul, Cristobal, Luca, Laura, Laurence, Loredana, Jonas, Floriane, Maude, Philemon, Lilith, Paula, Mihaly, Louis, Caroline. To my FdV mates: Roberta, Juanma, Miza, Guillermo, Aamir and others. To other Ph.D. students of the unit I got along with: Peter, Jo, Hector and Benoît.

This work would never be possible without help and patience of my family, my partner Arnaud and his family. Especially, I would like to thank Arnaud, who managed to be with me on the daily basis, spent the endless hours correcting my writing and speaking, discussed about the code and good practices, was making me laugh when I was coming home tired, angry or unmotivated, and just for being him adorable self. This thesis is dedicated to his father that will not be able anymore to profit from any breakthrough in cancer research.

I am proud to finish the thesis and face new professional adventures. I learned a lot about myself during this three years. I would like to thank very much everyone whom I crossed on this path. I hope to meet you again one day.

Motto

And now, let's repeat the Non-Conformist Oath!
I promise to be different!
I promise to be unique!
I promise not to repeat things other people say!
— Steve Martin, *A Wild and Crazy Guy* (1978)

Information

This thesis is also available online as a webpage at the address <http://urszulaczerwinska.github.io/UCzPhDThesis>. The web version contains interactive content (videos, figures, tables) that are static in the pdf version. Enjoy!

Contents

Preamble about Interdisciplinary Research	17
What does interdisciplinarity in science mean in XXI century?	18
Strengths, Weaknesses Opportunities, Threats (SWOT) of an interdisciplinary Ph.D. - personal perspective	20
The origins of the Ph.D. topic	22
Organisation of the dissertation	25
I Introduction	27
1 Immuno-biology of cancer	29
1.1 Cancer disease	29
1.1.1 Historical understanding of cancer	29
1.1.2 Tumor Microenvironment as a complex system	32
1.1.2.1 Interactions between TME and Tumor	33
1.1.2.2 Two-faced nature of immune cells: context-dependent functional plasticity	37
1.1.2.3 Immune cell (sub)types in TME	37
1.1.2.4 Summary	39
1.2 Quantifying and qualifying immune infiltration (data)	40
1.2.1 Cell sorting	40
1.2.1.1 Flow cytometry	40
1.2.1.2 Mass cytometry	41
1.2.2 Microscope Staining	41

1.2.2.1	Tissue Microarrays	41
1.2.3	omics	42
1.2.3.1	Transcriptome	42
1.2.3.2	Single cell RNA-seq	44
1.2.3.3	Epigenome	45
1.2.3.4	Copy number variation (CNV) and Copy number aberration (CNA)	45
1.2.3.5	Spatial transcriptomics	45
1.3	From cancer phenotyping to immune therapies	46
1.3.1	Cancer immune phenotypes	46
1.3.2	Scoring the immune infiltration	47
1.3.2.1	Immunoscore	47
1.3.2.2	Spatiotemporal dynamics of Intratumoral Immune Cells of Colorectal Cancer	49
1.3.2.3	Immunophenoscore	49
1.3.2.4	The immune landscape of cancer	50
1.3.2.5	A pan-cancer landscape of immune-cancer interactions in solid tumors	51
1.3.2.6	Immune maps	51
1.3.2.7	Summary	52
1.3.3	Immune signatures - biological perspective	52
1.3.4	Cancer therapies	54
1.3.5	Recent progress in immuno-therapies	54
1.4	Summary of the chapter	56

2	Mathematical foundation of cell-type deconvolution of biological data	59
2.1	Introduction to supervised and unsupervised learning	59
2.1.1	Supervised learning	60
2.1.2	Unsupervised learning	60
2.1.3	Low-dimensional embedding for visualization	61
2.2	Types of deconvolution	61
2.3	Cell-type deconvolution of bulk transcriptomes	63

2.3.1	Literature overview	65
2.3.1.1	Availability	65
2.3.1.2	Data type	68
2.3.1.3	Objectives of the cell-type deconvolution	68
2.3.1.4	Differences between approaches	68
2.3.1.5	Computational efficiency	69
2.3.2	Regression-based methods	69
2.3.3	Enrichment-based methods	74
2.3.4	Probabilistic methods	75
2.3.5	Convex-hull based methods	76
2.3.6	Matrix factorization methods	79
2.3.6.1	Principal Components Analysis	79
2.3.6.2	Non-negative matrix factorisation	82
2.3.6.3	Independent Components Analysis	84
2.3.7	Attractor metagenes	86
2.3.8	Others aspects	88
2.3.8.1	Types of biological reference	88
2.3.8.2	Data processing	89
2.3.8.3	Validation	90
2.3.8.4	Statistical significance	92
2.3.9	Summary	92
2.4	Deconvolution of other data types	93
2.4.1	DNA methylation data	93
2.4.2	Copy number aberrations (CNA)	94
2.5	Summary of the chapter	95
Objectives		97
II Results		99
3 Determining the optimal number of independent components for reproducible transcriptomic data analysis		101
3.1	Context	101

3.2 Description	102
3.3 Impact on the further work	102
4 Application of Independent Component Analysis to Tumor Transcriptomes Reveals Specific and Reproducible Immune-Related Signals	117
4.1 Context	117
4.2 Description	118
4.3 Impact on the further work	118
5 Comparison of reproducibility between NMF and ICA	133
5.1 Comparing metagenes obtained with NMF versus ICA	133
5.2 Summary	136
6 DeconICA: an R package for Deconvolution of omic data through Immune Components Analysis	137
6.1 From blind deconvolution to cell-type quantification: general overview	137
6.2 Unsupervised deconvolution	138
6.2.1 FastICA overdecomposition protocol	138
6.2.1.1 Data transformation	138
6.2.1.2 Determining k number of sources	138
6.2.1.3 FastICA	138
6.2.1.4 Orienting the components	140
6.3 Interpretation of the components	142
6.3.0.1 Identification of immune cell types with an enrichment test	142
6.3.0.2 Correlation based identification	143
6.3.0.3 Reciprocal match	143
6.3.0.4 Maximal correlation match	143
6.4 Computing the abundance of the identified cell-types	144
6.4.0.1 Defining markers	145
6.4.0.2 Computing scores	145
6.5 Validation of the abundance estimation with <i>DeconICA</i>	145

6.5.1 <i>In silico</i>	145
6.5.2 <i>In vitro</i>	145
6.5.3 PBMC transcriptome	147
6.6 Summary	147
7 Comparative analysis of cancer immune infiltration	153
7.1 Background	153
7.2 Methods	153
7.2.1 Data sources	153
7.2.2 The DeconICA pipeline on bulk	153
7.2.3 The DeconICA pipeline on single cell	154
7.3 Results	154
7.4 Discussion	154
7.5 Conclusions	154
7.6 Supplementary	154
7.6.1 Tables	154
8 A multiscale signalling network map of innate immune response in cancer reveals signatures of cell heterogeneity and functional polarization	163
8.1 Context	163
8.2 Description	164
8.3 Discussion and perspectives	165
III Discussion	209
9 Discussion	211
10 Conclusions and perspectives	217
10.1 Conclusions	217
10.2 Perspectives	218
Annexes	221
1 DeconICA documentation	221

1.1 Introduction to deconICA	221
1.2 Running fastICA with icasso stabilisation	262
1.3 Reference manual	271
2 Publications and conferences	309
2.1 Adjustment of dendritic cells to the breast-cancer mi- croenvironment is subset-specific	309
2.2 The inconvenience of data of convenience: computa- tional research beyond post-mortem analyses	325
2.3 CV: publications, conferences, courses	328
Glossary	333
Biological terms	333
Mathematical terms	333
Post Scriptum: Thesis writing	335
Bibliography	335

List of Tables

3	SWOT analysis of Interdisciplinary research	23
1.1	Six immunological subtypes of cancer	50
2.1	Summary of methods for cell-type deconvolution of bulk transcriptome	67
2.2	Contangency table	74
7.1	List of datasets	154
7.2	List of datasets 2	162

List of Figures

1	Symbolic illustration of a sum (multidisciplinarity) versus synergy (interdisciplinarity)	19
2	Interdisciplinarity of different fields.	21
1.1	Illustration of Virchow's cell theory	31
1.2	Percentage of publications containing the phrase "tumor immunotherapy" is growing	33
1.3	The microenvironment supports metastatic dissemination and colonization at secondary sites.	35
1.4	From Data to Wisdom	40
1.5	Five categories of RNA-seq data analysis.	43
1.6	Cancer-immune phenotypes: the immune-desert phenotype, the immune-excluded phenotype and the inflamed phenotype.	48
1.7	This timeline describes short history of FDA approval of checkpoint blocking immunotherapies up to 2017.	56
2.1	Illustration of the cocktail party problem	62
2.2	Principle of the deconvolution applied to transcriptome . . .	64
2.3	Distribution of publications of cell-type deconvolution of bulk transcriptome over the years	70
2.4	Simple statistics illustrating characteristics of published cell-type deconvolution tools	70
2.5	Principle of the SVR regression	72
2.6	Convex hull illustration	77

2.7	Fitting gene expression data of mixed populations to a convex hull shape	78
2.8	Principle of matrix factorisation of gene expression	80
2.9	Simple illustration of matrix factorisation methods	86
2.10	From theory to practice: simplified pipeline of model validation	91
5.1	Correlation graph of ICA and NMF multiple decompositions .	135
6.1	Flowchart of DeconICA method	139
6.2	Principle of components orienting	141
6.3	Example of sucessfull and unsucessful component matching to refrence	144
6.4	Accuracy of estimation versus true proporitons in an in silico mixture	146
6.5	Accuracy of estimation versus true proporitons in an in vitro mixture	146
6.6	Correlation between independent components and reference immune cell-type metagenes	148
6.7	Estimation of abundance of immune cell types in PBMC transcriptome of 104 healthy donors	149
6.8	Comparison of markers used by different deconvolution methods	150

Abbreviations

ACSN	Atlas of Cancer Signaling Networks
AI	A rtificial I ntelligence
BIODICA	ICA applied to B ig O mics D ata
BSS	B lind S ource S eparation
CAF	C ancer- A sociated F ibroblasts
CNA	C opy N umber A lterations
CNV	C opy N umber V ariation
CRI	C enter for I nterdisciplinary R esearch
DeconICA	D econvolution of omic data through I mmune C omponents A nalysis
DEG	D ifferentially E xpressed G enes
DGE	D ifferential G ene E xpression
EM	E xtracellular M atrix
EWAS	E pigenome- W ide a ssociation s tudy
FACS	F luorescence- a ctivated c ell s orting
GSEA	G ene S et E nrichment A nalysis
ICA	I ndependent C omponents A nalysis
ML	M achine L earning
mRNA	m essenger R NA
MSTD	M ost S table T ranscriptomic D imension
NGS	N ew G eneration S equencing
NK	N atural K iller
NMF	N on-negative M atrix F actorisation
PBMC	P eripheral b lood m ononuclear c ell
PCA	P rincipal C omponents A nalysis
RNA-seq	R NA s equencing
scRNA-seq	single c ell R NA s equencing
SVM	S upport V ector M achine
SVR	S upport V ector R egression
TCGA	T he C ancer G enome A tlas
TIL	T umor I nfiltrating L eucocytes
TMA	T issue M icroarrays
TME	T umor M icroenvironment
TPM	T ranscripts P er K ilobase M illion
t-SNE	T -distributed S tochastic N eighbor E mbedding
UMAP	U niform M anifold A pproximation and P rojection for Dimension Reduction

TCGA Study Abbreviations

Study Abbreviation	Study Name
LAML	Acute Myeloid Leukemia
ACC	Adrenocortical carcinoma
BLCA	Bladder Urothelial Carcinoma
LGG	Brain Lower Grade Glioma
BRCA	Breast invasive carcinoma
CESC	Cervical squamous cell carcinoma and endocervical adenocarcinoma
CHOL	Cholangiocarcinoma
LCML	Chronic Myelogenous Leukemia
COAD	Colon adenocarcinoma
CNTL	Controls
ESCA	Esophageal carcinoma
FPPP	FFPE Pilot Phase II
GBM	Glioblastoma multiforme
HNSC	Head and Neck squamous cell carcinoma
KICH	Kidney Chromophobe
KIRC	Kidney renal clear cell carcinoma
KIRP	Kidney renal papillary cell carcinoma
LIHC	Liver hepatocellular carcinoma
LUAD	Lung adenocarcinoma
LUSC	Lung squamous cell carcinoma
DLBC	Lymphoid Neoplasm Diffuse Large B-cell Lymphoma
MESO	Mesothelioma
MISC	Miscellaneous
OV	Ovarian serous cystadenocarcinoma
PAAD	Pancreatic adenocarcinoma
PCPG	Pheochromocytoma and Paraganglioma
PRAD	Prostate adenocarcinoma
READ	Rectum adenocarcinoma
SARC	Sarcoma
SKCM	Skin Cutaneous Melanoma
STAD	Stomach adenocarcinoma
TGCT	Testicular Germ Cell Tumors
THYM	Thymoma
THCA	Thyroid carcinoma
UCS	Uterine Carcinosarcoma
UCEC	Uterine Corpus Endometrial Carcinoma
UVM	Uveal Melanoma

Preamble about Interdisciplinary Research

We are not students of some subject matter, but students of problems. And problems may cut right across the borders of any subject matter or discipline. — Karl Popper

The piece of work you are reading should harvest the fruit of interdisciplinary research conceived in an interdisciplinary environment of Center for Interdisciplinary Research in Paris (CRI) in École doctorale *Frontières du Vivant* (FdV) and Institut Curie in groups Computational Systems Biology of Cancer and Integrative Biology of Human Dendritic Cells and T-cells. CRI's main mission can be formulated as follows:

*to empower the students to take initiative and develop their own research projects at the **crossroads of life, learning, and digital sciences.** [?]*

Interdisciplinarity has many definitions and meanings. According to the book *Facilitating Interdisciplinary Research* [?]

*Interdisciplinary research and education are inspired by the drive to solve **complex questions** and problems, whether generated by scientific curiosity or by society, and lead researchers in different disciplines to meet at the **interfaces** and **frontiers** of those disciplines and even to **cross frontiers** to **form new disciplines.***

For me, the essence of interdisciplinarity is the need to solve a complex problem, whatever expertise would be necessary to solve it. I consider that fighting cancer disease, deciphering cancer heterogeneity and interactions of the immune system are causes worth an interdisciplinary effort. This is even truer in the era of big data when the demand for quantitative tools is exponentially growing, in order to extract information and knowledge.

Though this preamble I would like praise not only the interdisciplinary research but also underline possible limitations and constraints that come with it and which could affect this thesis.

What does interdisciplinarity in science mean in XXI century?

In the ancient history, being formed and practice multiple disciplines was not anything unusual which is strongly reflected in Greek philosophy initiating the dispute about the division and hierarchical classification of knowledge. [?]. Figures as Aristotle and Leonardo Da Vinci that can be called *homo universals* served different disciplines from arts through history, natural sciences to mathematics. With time human knowledge about the world, i.e., natural sciences got bigger and bigger, to the point that it became hard to master all the disciplines. The specialization would allow to study in deep a certain subject and make possible discoveries about it. And even if, interdisciplinary efforts never stopped, for a long time they were not mainstream in scientific communities divided into academies, chairs, and specialization.

Different fields differ in term of concept, method, tools, processes, and theories [?]. Thanks to division into scientific disciplines a sort of order is conserved across space and time. Hierarchical classification of knowledge comes from human nature.

It can be observed that there is an increasing gap between disciplines along with specialization.

advancing specialisation leads to gaps in the level of comprehension between individual disciplines and eventually gives rise to the demand for interdisciplinarity - in order to close the gaps between disciplines.[?]

It is not really clear why this gap must happen. Would it somehow reflect human nature, the strong need to divide things into discrete categories rather than to see a continuum?

Nowadays, the knowledge is accessible, and we can profit from achievements of different disciplines thanks to easy means of communication. Two different terms can be defined to describe initiatives that use the knowledge of different specialties: multidisciplinarity which is a sum of efforts of different disciplines and interdisciplinarity that allows profiting from the synergy of multiple disciplines (Fig. 1). With interdisciplinary research and education come flexibility, creativity, and novelty but also limit of depth on ingested knowledge and possibilities of cross-interactions between disciplines.

Why are not all of the labs interdisciplinary?

Scientists tend to resist interdisciplinary inquiries into their own territory. In many instances, such parochialism is founded on the fear that intrusion from other disciplines would compete unfairly for limited financial resources and thus diminish their own opportunity for research — Hannes Alfvén

Crossing frontiers is not an easy task, and it was quite difficult in the beginnings of modern interdisciplinarity. Some examples of early interdisciplinary efforts of the 20th century are nicely described by Ledford et al. [?] in *Nature* special issue on [Interdisciplinarity](#). It illustrates Theodore

Multidisciplinarity	<	Interdisciplinarity
A + B + C	<	A + B + C
A + B	<	A + B
B + C	<	B + C
C + A	<	C + A

Simple sum of disciplines Synergy effect

Disciplines: A; B; C;

Figure 1: Symbolic illustration of a sum (multidisciplinarity) versus synergy (interdisciplinarity), in an interdisciplinary project sum of three disciplines A, B, C should have more value than a simple sum of disciplines: an interdisciplinary project should have an added value compared to a multidisciplinary one.

Brown in 1980s while trying to organize a new interdisciplinary research project and reorganize university space to engage an exchange between students of different faculties, and he encounters a lot of reluctance.

And then there was the stigma. "Interdisciplinary research is for people who aren't good enough to make it in their own field," an illustrious physicist chided [?].

The story seems to end up with a happy ending of 40-million US dollars grant and foundation of Beckman Institute for Advanced Science and Technology. However, recruiting an open-minded director for leading this unconventional organization was a struggle. Shortly, the structure became a model for others and met a great scientific and technological success.

Even though, since then the idea of interdisciplinary research spread around the world. Yet, not all problems were overcome.

"There's a huge push to call your work interdisciplinary," says David Wood, a bio-engineer at the University of Minnesota in Minneapolis. "But there's still resistance to doing actual interdisciplinary science".

First, the institutions, universities where research is performed should equip scientist with a passport to other disciplines, facilitate exchange, funding the interdisciplinary research, be accepting fusion of disciplines as new ones. Then, proper communication between disciplines is necessary. Finally, developing interdisciplinary research is extremely challenging as it often requires extra effort from an apprentice.

Are all the disciplines independent units nowadays?

Can we do molecular biology without technical, mathematical and computational support? Can we study cognitive science without knowledge of biology, physics, and psychology? Can we advance medicine without basic research in biology, physiology, electronics?

Bioinformatics and/or computational biology is an compelling case. Working in this field is being between biology, medicine, computer science, mathematics and statistics, the role of a computational biologist is sometimes reduced to a service. A biological lab may need a computational biologist to perform an analysis, restructure the data, that is needed for the biological discovery. Often, there is not enough space for research in computational biology itself, where the discovery does not depend on the original data but tools and approaches to complex, data-intensive biological problems. It may also happen the other way round when a computational biologist asks a bench researcher to perform an experiment to prove his theoretical model. In both cases, the long-term interdisciplinary partnership would probably fail. Wet and dry researchers should collaborate as equal with important research advances on both sides to assure a long-term equilibrium.

How did interdisciplinarity change over the years? Are all disciplines affected equally?

From the chart (Fig. 2), we can notice that Social Studies of Medicine seems to be the most interdisciplinary field. In general Biology, Health and Biomedical Sciences seem to be more open into a flow of knowledge from other fields than humanities. On the extreme opposite of health, Clinical Medicine appears to be a very conservative field.

Strengths, Weaknesses Opportunities, Threats (SWOT) of an interdisciplinary Ph.D. - personal perspective

I'm not good enough to do well something I dislike. In fact, I find it hard enough to do well something that I like — Jim Watson, Succeeding In Science: Some Rules Of Thumb [?]

Being formed first in a double major in biology and mathematics, then participating in interdisciplinary research projects during my master studies, I can witness that the learning curve of multiple disciplines can be steep. It is also often associated with the frustration of not going deep enough in all of the disciplines or the feeling of being overwhelmed by the amount of knowledge.

Coming with the expertise in biology and mathematics, I got fascinated by complex biological systems. One way of study high-dimensional data is to reduce them into smaller interpretable units. This is what I tempted to achieve in this thesis in order to enrich our knowledge about tumor microenvironment and possibly contribute to orienting future research on immunotherapies.

However, being an interdisciplinary researcher was not always a privilege. *To which category do I belong? To whom should I present my work?* I often asked myself these questions. I also

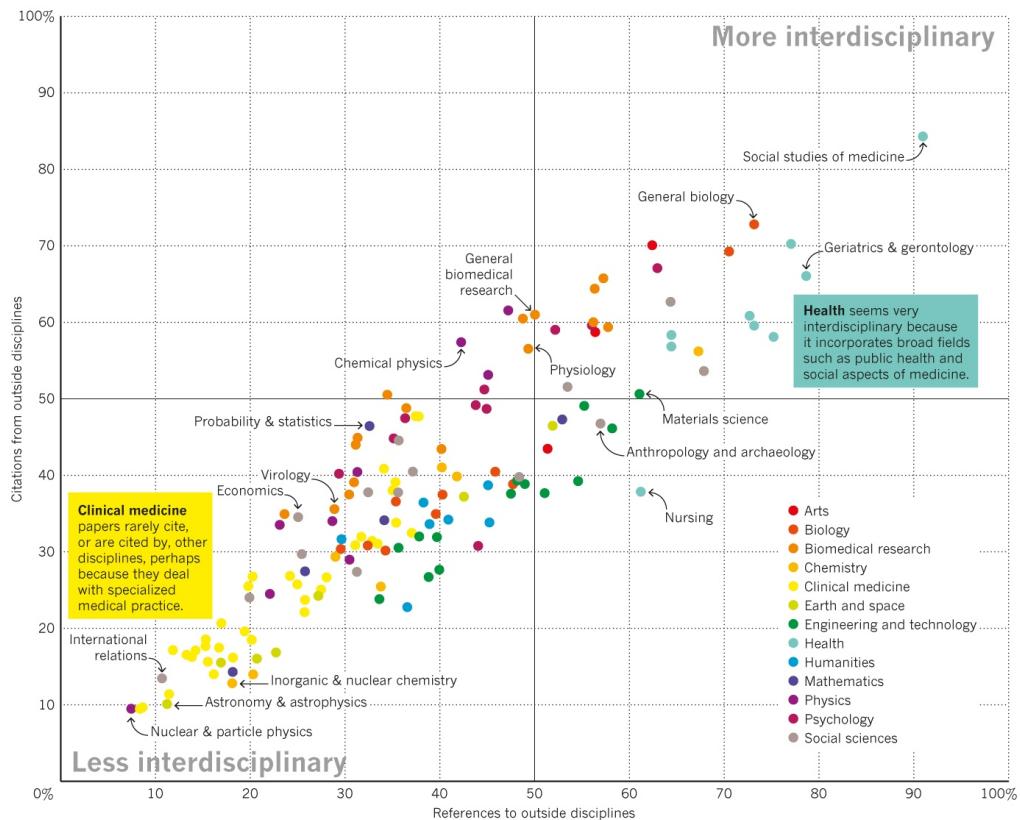


Figure 2: Interdisciplinarity of different fields. “From 1950-2014, a field’s position is determined by how much its papers cite outside disciplines (x-axis), and by how much outside disciplines subsequently cite its papers (y-axis). (Some years, certain fields have too few references to be plotted.)”. Reprinted by permission from Springer Nature [?] © 2015 Nature America, Inc. All rights reserved.

often encountered lack of understanding where my methodological results were not bringing enough of *biological insights*. Or the constraints of my biological application seemed very obscured and complicated for mathematicians, and my work often lacked *important methodological advances*.

Does it mean that my work is not accurate, useless? Probably, for many, it is not enough. However, I still hope that our findings will be interesting to some. I enjoy working with data and statistics that serve an actual purpose. The Tab. 3 summarizes Strengths, Weaknesses, Opportunities, and Threats (SWOT analysis) of an interdisciplinary project, in the way I perceive it.

Besides conducting research that crosses the boundaries of one discipline, I also could meet and work with inspiring people coping like me with filling the gap in understanding of interdisciplinary work, multiple supervisors and report to many institutions. I gained (even if only superficial) understanding of many topics in mathematics, statistics, data science, immunology, cancer but also oral and written presentation skills, time and work management

Is my thesis genuinely interdisciplinary? Does biology profits from mathematics and mathematics from biology? I will let you judge it.

What impact had biology on the statistical/mathematical modeling? The practical problems, systems that go beyond theoretical formulations challenge the theoretical tools. In my work, I did my best to fuse theory and practice that should serve a biological application. I can image the project more complete if the results of my work would inspire changes in biological experiments, uncover new paths to follow for experimental biologists or translational researchers.

The origins of the Ph.D. topic

The universe will lead me where I need to go. I am like a leaf in the stream of creation — Dirk Gently, Holistic detective

When finishing my master, I was looking for an interdisciplinary topic where I could deepen my quantitative skills and apply to a real-life healthcare problem. I came across a project proposed by Andrei Zinovyev in close collaboration with Vassili Soumelis. I was quite anxious that my knowledge of cancer immunology would not be sufficient to lead the project to a success. I recognize that the immune systems are very complex and dynamic system and many years of expertise are needed to grasp an understanding of it really. I had a great chance to work hand in hand with domain experts that would suggest me the direction I should take in my research.

The project started by causal exploration of different blind source separation or dimension reduction techniques and their ability to dissect bulk transcriptomic data into cell type-related units. We also faced a vital problem of lack of gold standard validation data that would define efficiency and accuracy of different methods.

I have spent fruitless efforts working on a bulk transcriptomic data simulation framework, important statistical issues come our way and probably another few years of a different Ph.D. would be

Table 3: SWOT analysis of Interdisciplinary research. In SWOT analysis, Strengths, Weaknesses, Opportunities, and Threats are enumerated. Strengths and Weaknesses are internal, and Opportunities and Threats are external factors.

Strengths (internal, positive)	Weaknesses (internal, negative)	Opportunities (external, positive)	Threats (external, negative)
Having a holistic view of the problem	Not seeing details of the problem	Mulitple possibilities to convey research	Spending too much time filling knowledge gap
Being supervised by multiple experts	Following multiple, sometimes contradictory, advice on the same problem	Take advantage of synergistic effect of fields	Inhibiting effect of oppinions from different fields
Joining expertises of different fields	Not covering in details all the disciplines	Doing a new discovery	Obtaining too generic results
Using new/non standard approach	Experiencing steep learning curve	Raising interest in different expert domains	Not mastering the specific vocabulary of different fields
Having better understanding of complex processes	Being in constant need of help of domain experts	Making progress	Not being understood
Higher creativity		Creating a new field	Being hard to classify/ fall into a category
Having great flexibility		Sovling many problems impossible to solve with traditional approach	Being considered as superficial
Feeling a thrill of adventure Being open			

necessary to solve them. In the meantime, many tools dissecting tumor bulk transcriptome were published. Serving a similar purpose, they used different means and assumptions, which left a space for my project to continue. In my third year, I am finally publishing a tool that performs the analysis I developed together with the Sysbio team members, and I can apply it to a corpus of publicly available data to learn about the actual question: the immune system infiltrating cancers and the context-dependent signatures (see Chapters 4 & 5).

In a parallel project, I worked on an exploration of a brand new data type: single cell transcriptomic (RNAseq) in the context of tumor microenvironment (see Chapter 6).

We have also participated in the Dream Idea Challenge, a project that aimed to put closer experimental and theoretical researchers (Annexe 1, [?]).

I have collaborated in numerous projects within and outside my team. Some of the projects resulted in publications, such as my work on analyzing pDC subsets of breast cancer Annexe2. Some others are in still preparation.

I have attended nine national, and international conferences, where I presented posters, gave talks and I got awarded with distinctions for my work.

Alongside with pursuing the compelling scientific research, I completed a wide variety of courses and I was teaching IT, Statistics and Mathematics at pharmacology faculty. Thanks to this extensive (>300 hours) training over three years, I am equipped with soft skills that not only helped me to shape my thesis project on the go but also, I hope, will help me to succeed in my future career path.

Organisation of the dissertation

As it is a fruit of an interdisciplinary work, I decided to introduce the topic from two perspectives: describe the biological and biomedical dimension of the topic (see Chapter 1), as well as, the mathematical dimension of the problem of separation of sources in complex mixtures (see Chapter 2). I hope, it will make the subject of my thesis easy to understand also for non-biologists or non-mathematicians. In the results part, I introduce a study of ICA applied to transcriptomes (Chapter 3). I also apply ICA-based deconvolution to Breast cancer transcriptomes to prove its reproducibility Chapter 4. I compare the reproducibility of blind source separation methods NMF and ICA (see Chapter 5). Then I introduce the DeconICA R package (see Chapter 6) and finally present results of an application of DeconICA and other tools to 118 transcriptomic datasets (see Chapter 7). The second part of the results is dedicated to my work on cell type heterogeneity (see Chapter 8). The manuscript finishes with Chapter 9 and Chapter 10 that contain discussion, conclusions, and perspectives. In annexes, you can find publications to which I contributed during my doctorate that are not strictly linked with the topic of this thesis. In the end, I included a glossary of useful terms.

INTRODUCTION

- Chapter 1: introduction to cancer biology and immunity, challenges in cancer immunotherapies and cancer immune phenotyping as well as data sources most commonly used to face the topic.
- Chapter 2: introduction to a problem of mixed sources in biological samples, an overview of blind source separation methods and supervised deconvolution methods, with focus on those applied to bulk transcriptome to uncover and quantify immune compartments

RESULTS

- Chapter 3: Most Reproducible Transcriptome Dimension (MSTD)
- Chapter 4: application of ICA-based deconvolution to six breast transcriptomes
- Chapter 5: comparison of reproducibility of NMF and ICA methods
- Chapter 6: DeconICA R package
- Chapter 7: application of DeconICA R package and other tools to analyze >100 transcriptome datasets of bulk cancer transcriptomes

- Chapter 8: study of immune cell types heterogeneity in tumor microenvironment using the innate immune map and scRNA-seq data

DISCUSSION

- Chapter 9: Discussion
- Chapter 10: Conclusions and perspectives

ANNEXES

- Other publications:
 - Adjustment of dendritic cells to the breast-cancer microenvironment is subset specific
 - The inconvenience of data of convenience: computational research beyond post-mortem analyses
- DeconICA R package documentation:
 - Vignette 1: Introduction to deconICA
 - Vignette 2: Running fastICA with icasso stabilization
 - Manual
- Scientific CV (including a list of attended conferences and publications)

GLOSSARY

Part I

Introduction

Chapter 1

Immuno-biology of cancer

This chapter will first introduce a short history of cancer with a focus on discoveries linking cancer and its environment. It will also describe the participation of TME in cancer development, progression and response to treatment. Most important types of data used to study cancer microenvironment will be discussed. I also introduce a link between tumor immune-biology and cancer phenotyping for development of immunotherapies.

1.1 Cancer disease

According to [GLOBOCAN study](#) [?], 14.1 million cancer cases were estimated to happen around the world in 2012. It touched 7.4 million men and 6.7 million women. It is estimated that the cancer cases will increase almost two-fold to 24 million by 2035.

In France only, in 2012 there were 349426 cases of cancer, of which leading is Prostate cancer (16,3%) followed by Breast (14%) and Lung (11,5%).

For a long time studying tumor was focused on tumor cells, their reprogramming, mutations. Cancer was seen as a disease of uncontrolled cells by the mainstream research. At the same time, the idea of the importance of the impact of other cells and structures on cancer cells was present but often not believed. A recent success of immunotherapies moved research focus to tumor cells in their context: tumor microenvironment. We will describe here what is the composition and role of the TME in tumor progression, diagnosis and response to treatment.

1.1.1 Historical understanding of cancer

Cancer was historically described by a physician Hippocrates (460–370 B.C) [?]. Even though there exist even earlier evidence of the disease. Hippocrates stated that the body contained 4 humors (body fluids): blood, phlegm, yellow bile and black bile. Any imbalance of these fluids will

result in disease. Particularly the excess of black bile in an organ was meant to provoke cancer. For years, it was not known what factors cause cancer and it was easily confounded with other diseases. In the middle ages in the Renaissance Period, it was believed cancer is a punishment for the sins they committed against their god, that they deserved it to some extent.

Until the 18th century, it was believed that cancer is contagious and is spread by parasites.

In the 19th century, tumor cells started to be analyzed by pathologists. They were struck with their ability to proliferate uncontrollably, ability to spread and destroy the original tissue [?]. Around the same time, leukocytes from the blood were first described by Gabriel Andra and William Addison. Just a few years later, in 1845 Bennett and Virchow described blood cells in leukemia (Fig. 1.1). Virchow is also a father of Chronic irritation theory (nowadays called chronic inflammation) that says that cancer is caused by local “irritation” and, incorrectly, that cancer cells spread like liquid resulting in metastasis.

In 1889, Stephen Paget introduced *soil and seed* hypothesis of metastases [?]. He formulates it as follows

When a plant goes to seed, its seeds are carried in all directions, but they can only live and grow if they fall on congenial soil.

Which is parallel to cancer cells disseminated by body fluids, and they can grow only in tissues - “soil” that is predisposed to host the cancer cell - “the seed”. He focused on the importance of tissue characteristics that favorize tumor development as opposed to most researchers of his time that were focusing on the “seed” itself.

In the 20th century, molecular causes started to be investigated. It was discovered that cancer could be caused by environmental factors, i.e. chemicals (carcinogens), radiation, viruses and also inherited from ancestors. Those factors would damage but contrary to a healthy condition they would not die.

Also in 1909, Paul Ehrlich, called one of fathers of immunology and Nobel Prize laureate, indicated a link between immune system and tumor suppression [?]. One of the remarkable first immunotherapy attempts can be attributed to William Coley, that practiced injecting streptococcus bacteria directly into patients after cancer surgery in 1891, later called “Colley vaccine”. However, the impact of this procedure on patients recovery was judged by scientific community as “unclear”.

In 1968, Melvin Greenblatt and Philippe Shubik showed that tumor transplants secrete a substance stimulating the growth of blood vessels [?], later identified as “tumor angiogenic factor (TAF)” by Judah Folkman in 1971 [?]. Folkman also suggested that TAF can be a target of a therapy itself. This was a revolutionary idea, at the time, as it did not target the tumor cells directly but acted on their environment.

During the 1970s, oncogenes and tumor suppressor genes were discovered. Oncogenes are genes that allow a cell to become a cancer cell, while the tumor suppressor genes would repair DNA or execute cell death of a damaged cell. A new dimension to cancer studies was added in

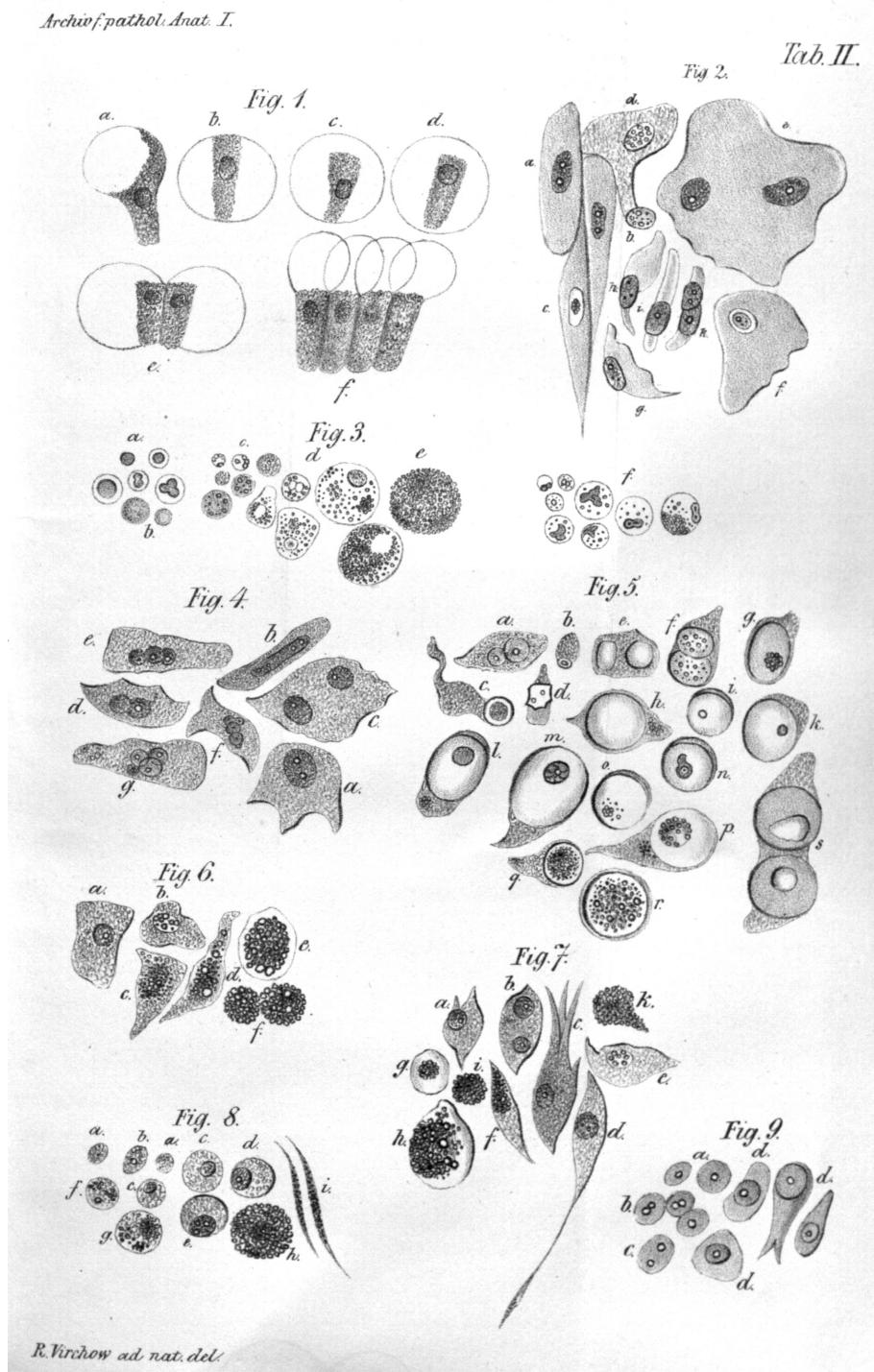


Figure 1.1: Illustration of Virchow's cell theory. Virchow depicted different cells transformation due to irritation. [?]

the 1980s, epigenetic changes were proven to occur to both oncogenes and tumor suppressors [? ?], which are presently known as epigenetic markers used for diagnostics and therapeutic targets for cancer.

In 1982, Aline van Pel and Thierry Boon [?] discovered that a specific immunity to spontaneous tumor cells could be induced by vaccinating mice with mutagenized tumor cells. This raised an inspiration for many years of immune therapy development.

In Napoleone Ferrara and colleagues identified the gene encoding vascular endothelial growth factor (VEGF) that was shown to stimulate the growth of endothelial cells proliferation *in vitro* and angiogenesis (blood vessels formation) *in vivo* [?].

In 1999 for the first time, gene-expression was used to study cancer (leukemia) by Todd Golub, Donna Slonim and colleagues [?].

Since the end of the 20th century, cancer screens are developed along with multiple strategies to fight the tumor. Most classical ones are based on the idea of removing tumor cells (surgery), killing tumor cells with DNA-blocking drugs (chemotherapy), radiation, inhibit cancer growth (hormonal therapy, adjuvant therapy and immunotherapy). As none of those methods is fully efficient, often a combination of treatments is proposed. Nowadays, science is aiming in the direction of targeted therapies and personalized treatment.

The recent success of immunotherapies (discussed in Immunotherapies section attracted the attention the scientific community again to the context in which tumor cells are found. This context called Tumor Microenvironment, as well as the communication that happens within it between different agents nowadays studied differently with available knowledge of molecular biology, have become a popular scientific topic of the 21st century (Fig. 1.2).

1.1.2 Tumor Microenvironment as a complex system

Tumor Microenvironment is a complex tissue that surrounds tumor cells. It is composed of different compartments (in solid tumors):

- Stroma: blood and lymphatics vessels, epithelial cells, mesenchymal stem cells, fibroblasts, adipocytes supported by extracellular matrix (EM)
- Immune cells: T cells, B cells, NK cells, Dendritic cells, Macrophages, Monocytes etc.

Their proportion and specific roles vary significantly with tumor type and stage. Communication between the environmental cells and the tumor is critical for tumor development and has an impact on patient's response to treatment. This communication between different compartments is bidirectional and all the players can influence each other. Depending on the nature and prevailing direction of those interactions different destiny is possible for each of the compartments, i.e. immune cells can be recruited to protect tumor cells or they can kill them directly. Many of the signals can be contradictory, many can suppress each other. Then is it possible to tilt this

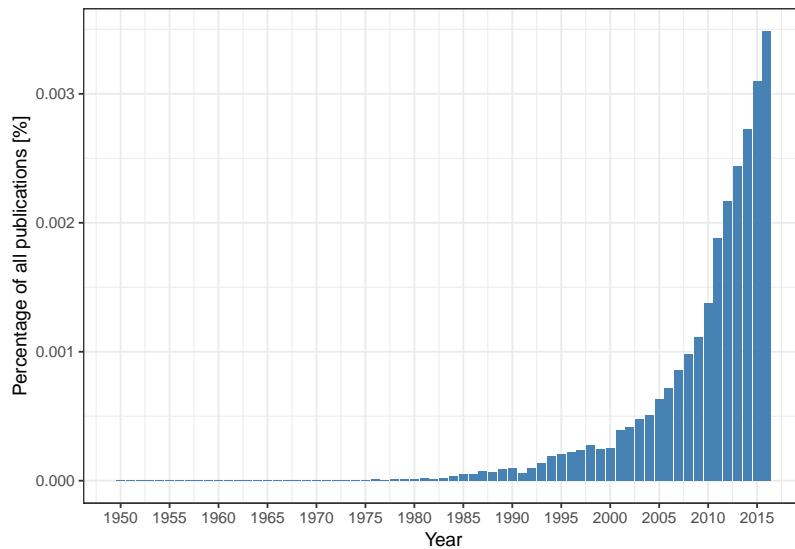


Figure 1.2: Percentage of publications containing the phrase “tumor immunotherapy” is growing, numbers retrieved on 17.01.2018 from [Medline Trends](#) [?]

complex ecosystem into patients’ favor? Can we decipher the most important factors of this molecular knot and manipulate it?

Next section describes different scenarios of interaction within TME in order to illustrate the complexity of TME and possible targets for cancer therapies.

1.1.2.1 Interactions between TME and Tumor

Three scenarios can be considered to describe the relationship between TME and tumor cells:

1. TME stimulates tumor growth and/or progression and/or impact negatively the response to treatment
2. TME has no influence on tumor cells and disease development
3. TME has a tumor-suppressive role and impact positively the response to treatment

As it is presented in Section 1.1.1 these three hypotheses were gaining and losing popularity in the scientific and medical community over the decades.

1.1.2.1.1 TME as a foe: inflammation

In 1863 Rudolf Virchow observed a link between chronic inflammation and tumorigenesis. According to Virchow theory, the genetic damage would be the “match that lights the fire” of cancer, and the inflammation or cytokines produced by immune cells should be the “fuel that feeds

the flames" [?]. Therefore lymphocyte infiltration was confirmed by subsequent studies as a hallmark of cancer. The question one may ask is why our immune system is not enough to defend the organism from tumor cells as it does efficiently in a range of bacterial and viral infections? It is mainly because of the ability of tumor cells to inhibit immune response through activation of negative regulatory pathways (so-called immune checkpoints).

It is worth mentioning, that the immune system can be already disabled and therefore cancer has more facility to develop. The immune system can be less efficient for example because of drugs given to patients after transplants or because of diseases like HIV/AIDS. These people have higher probability to develop cancers caused by infectious agents (viruses). These cancers are non-Hodgkin lymphoma (NHL) (caused by [Epstein-Barr virus](#) (EBV) infection), lung (no identified specific infectious agent), kidney (no identified specific infectious agent) and liver (caused by chronic infection with the [hepatitis B](#) (HBV) and [hepatitis C](#) (HCV) viruses) cancers. Human papillomavirus (HPV), can cause cervical, anal, oropharyngeal, and other cancers.

In the case of non-infectious cancers in patients with no history of immunosuppressive drugs intake or diseases, the question how tumor manages to break natural defence remains even more interesting. Many examples can be cited on how TME facilitates tumor development (Fig. 1.3). For instance, in the early stages of tumorigenesis, some macrophage phenotypes support tumor growth and mobility through TGF-beta signaling. Also, it was shown that NK cells and myeloid-derived suppressor cells (MDSCs) have an ability to suppress immune defence i.e. immuno-surveillance by dendritic cells (DCs), T cell activation and macrophage polarisation and they promote tumor vascularization as well. [? ?] They create so-called niches that facilitate tumor colonization. T-reg and myeloid-derived suppressor cells can negatively impact natural immune defense and by these means allow growth and invasion of tumor cells [?]. Another cell type, a part of EM, fibroblast, or more precisely Cancer-Associated Fibroblasts (CAFs) have proven pro-tumor functions in breast cancer where they enhance metastasis [?]. The blood and lymphatic vessels maintain tumor growth providing necessary nutritive compound to malignant cells.

According to [?] immune and stroma cells participate in almost all of Cancer Hallmarks [? ?]. Most of the hallmarks of cancer are enabled and sustained to varying degrees through contributions from repertoires of stromal cell types and distinctive subcell types.

1.1.2.1.2 TME seen as neutral

In front of lack of definitive proof that TME can positively or negatively impact on tumor development, many scientists, in a long time, ignored the importance of this factor. Until the early-mid eighties, the TME research was mostly limited to angiogenesis and immune environment and most areas that are now driving the field were not represented.

From the early 70s until the end of the 90s, the most accepted statement was that genetic alterations in oncogenes and tumor suppressor genes are both necessary and sufficient to initiate tumorigenesis and drive tumor progression. Therefore TME was not seen as an important element of the puzzle.

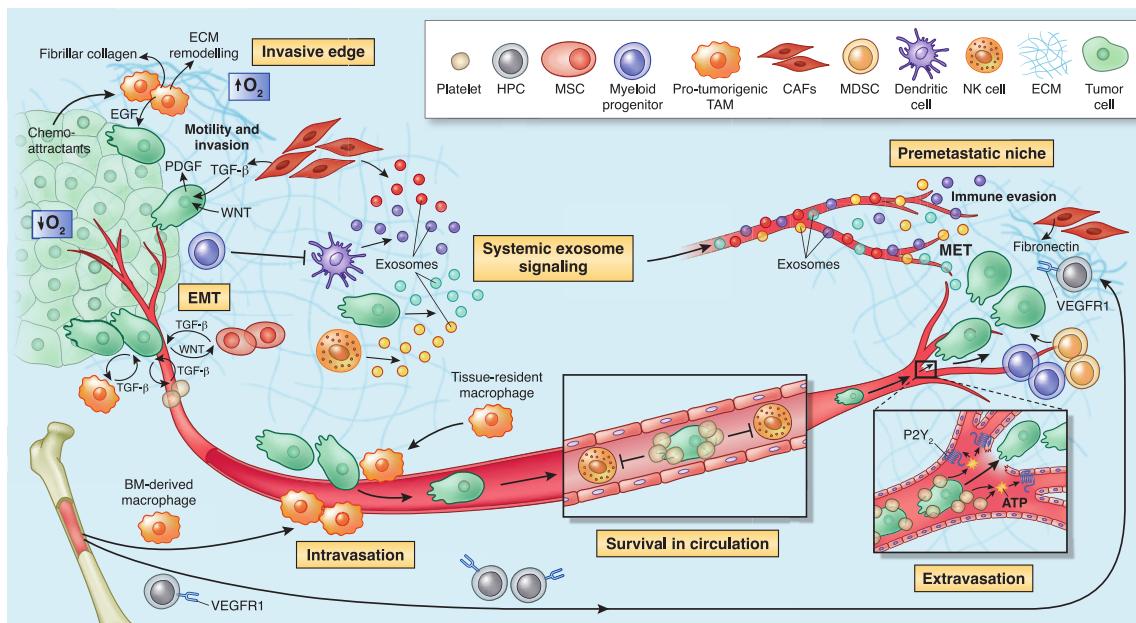


Figure 1.3: The microenvironment supports metastatic dissemination and colonization at secondary sites. Different tumor sites can communicate through exosomes realized by tumor cells and also immune and stromal cells such as NK cells, CAFs and DCs. Reprinted by permission from Springer Nature [?] © 2013 Nature America, Inc. All rights reserved.

The cancer geneticists, at the time, had a lot of influence on scientific community diminishing the work made on TME which were considered as “uninteresting” and definitely not “mainstream”.

After the 90s, with the discovery of signaling molecules involved in the communication of TME like VEGF general opinion started to change. Furthermore, discoveries made by developmental biology field supported the hypothesis that microenvironment plays an important role in development which was later shown for tumorigenesis. Additionally, the success of immune vaccines starting with the tuberculosis vaccine Bacille Calmette-Guérin (BCG) in 1976 and finishing, at the moment with checkpoint inhibitors did not leave the scientific community indifferent.

1.1.2.1.3 TME as a friend: immunosurveillance

As mentioned in Section 1.1.1 Paget proposed a hypothesis of “seed and soil” where the TME in a certain tissue (the soil) can either stimulate or suppress the metastasis (the seed). William Coley tested a possibility to trigger tumor-suppressive effect via stimulation of the immune system with bacteria. In the 1960s, the immune surveillance theory hypothesized “the ability to identify and destroy nascent tumors as a central asset of the immune system” [? ?]. Thus, the hypothesis that TME can have a positive role in tumor prognosis is not new.

In modern immuno-oncology, the term *immune-editing* was introduced by [?] in 2002, to describe the relationship between the tumor cells and the immune system. The immunosurveillance through immune-editing can be summarized in three processes: elimination, equilibrium, and escape [?].

The elimination is the direct killing of cancer cells or growth inhibition by the immune system. The adoptive T cells and NK are actively involved in tumor killing and stimulate other immune cells. The CD8 + cytotoxic lymphocytes (CTLs) directly recognize tumor cells. Employing perforin- and granzyme-dependent mechanisms they can lyse tumor cells. The CD4 + T cells release factors to induce proliferation of B cells and to promote their differentiation to the antibody (Ab)-secreting plasma cells, activate macrophages. Macrophages use phagocytosis to eliminate cancer cells [?].

The tumor-infiltrating lymphocytes (TILs) have been associated with an overall good prognosis and better survival in different cancer studies. Moreover, abundance of CD3 + and CD8 + T cells, NK cells, and $\gamma\delta$ T cells correlate with improved outcomes in epithelial ovarian cancers [?]. Several studies report that the presence of the abundant immune infiltrates is correlated with a good prognosis or better survival [? ? ? ?]. Spontaneous regression of human tumors has been reported in cutaneous melanoma, retinoblastoma, osteosarcoma, etc. [?].

The equilibrium is the phase when cancer and immune cells coexist and their crosstalk is preventing metastasis.

T cells are the main actor in maintaining the equilibrium. Progressively, the tumor cells become more immunogenic as they are not edited by the immune system [?]. The state of tumor cells is then identified as “dormant” and active scientific reports investigate the possible molecular pathways that maintain dormancy or lead to escape [?].

The immune escape is the final process when tumor cells impair the immune response.

1.1.2.2 Two-faced nature of immune cells: context-dependent functional plasticity

A modern vision of TME-tumor interactions assumes that tumor can be directed to several molecular pathways. This direction is decided by signals that are native of tumor cell and/or coming from the microenvironment.

Recent studies unveil ambivalent nature of immune cells in TME. While some as cytotoxic T cells, B cells and macrophages can manage to eliminate tumor cells. Treg cells role is to regulate expansion and activation of T and B cells. Depending on cancer type, they can be either pro- or anti-tumor. For example, as it has been shown for T-reg, that are usually associated with bad prognosis, they can be equally associated with improved survival (i.e. in colorectal cancer [?]). For innate immunity, there are widely accepted M1 (anti-tumor) and M2 (pro-tumor) extreme macrophages phenotypes in TME [?]. Most of the statements seem to be context dependent and not valid universally across all cancer types. We already mentioned Macrophages phenotypic plasticity as well as the different behavior of EMC depending on tumor stage.

From a more general point of view, it has been observed that immunodeficiency can correlate with high cancer incidence. Results of analysis based on observations of 25,914 female immuno-suppressed organ transplant recipients, the tumor incidence was higher than predicted for multiple cancers. However, the number of breast cancer cases decreased which can be really disturbing if we need to decide on the role of immune defense in tumor progression [?]. This indicates that immune microenvironment can be cancer stimulating or inhibiting depending on the type of cancer and/or other factors.

1.1.2.3 Immune cell (sub)types in TME

We are taught that a cell is the basic structural, functional, and biological unit of all known living organisms. A human body contains around 10^{14} which is three orders of magnitude more than the number of stars in the Milky Way. This ensemble of cells is traditionally classified into cell types based on their phenotypical variety.

for their immense number, the variety of cells is much smaller: only about 200 different cell types are represented in the collection of about 10^{14} cells that make up our bodies. These cells have diverse capabilities and, superficially, have remarkably different shapes.... ?]

In the description of TME, I have referred to cell types of immune cells as well-established entities of the immune system. However, the definition of cell types remains controversial and there is no consensus among researchers on how exactly a cell type should be defined. The notion of the cell-subtypes is even vaguer. The problem does not only concerns immune cells, most of the cell types of our organism, classified initially according to their morphology, seem to fulfill multiple

functions. One can also relate cell-type problem to species problem where scientist also debates about where to draw the borders between species. This problem is widely generalized as “theory of types” [?] in many disciplines as philosophy, linguistics, mathematics.

In this chapter, I will limit the description to immune cell types.

An immune cell can be described nowadays along many axes:

- Phenotype /surface markers
- Morphology (expressed proteins)
- Ultrastructure (electron microscopy)
- Molecular data (gene expression, genotype, epigenome)
- Cell fate
- Cell of origin
- Function

Depending on how well a cell is different from all other cells along with those axes, it will (or not) be defined as a distinct cell type. Each of these axes contains a piece of information that can agree with other axis or not. These features can be independent or can overlap depending on cell types in question. Historically, there were given different importance based on the technologies and general tendencies. Thus, there is no available general recipe applicable to discriminate all cell types from each other. Moreover, usually, it is not possible to measure all these axes simultaneously (because of the experimental, money or other constraints). Therefore, depending on the scientific question, different researches will give different weights to these axes and a combination of 2-3 *most important* features will be used to discriminate cell types in a given study.

Besides, the discrimination of the cell types comes with more or less subjective threshold on where the cells become *significantly different*. These thresholds can be established computationally or by an expert. The usual practice is a mix of both methods.

Since the beginning of immunology, there was disagreement between pre-defined cell types and cell functions.

Cette espèce de leucocytes a une grande ressemblance avec certains éléments fixes du tissu conjonctif, ainsi qu'avec des cellules endothéliales et des cellules de la pulpe splénique. On est donc souvent embarrassé, surtout lorsqu'on trouve ces leucocytes mononucléaires en dehors des vaisseaux, pour les distinguer des autres espèces de cellules mentionnées. — Elie Metchnikoff, Leçons sur la pathologie comparée de l'inflammation, 1891

The definition of cell types and subtypes is widely discussed today with the arrival of single cell technologies that allow a change of paradigm in cell classifications. Up to now, the top-down approach was mostly used. A pre-defined set of parameters describing a cell was fixed in order to select cells and then other parameters were measured. Now, it is possible to practice bottom-up approach where all (or some) parameters are measured for a single cell and then, depending on its distance from other cells, cell types are defined [?].

The concept of “cell type” is poorly defined and incredibly useful
— Alon Klein, Harvard Medical School

Researchers recognize that the concept of cell type is artificial and a continuum of cell types is closer to the reality. According to Susanne Rafelski,

A useful way to classify cells might thus be a multiscale and multi-parameter cell-type space that includes vectors for key intracellular organizational, dynamic, and functional features as well as tissue location, gene expression etc.

Some, as Alon Klein, propose to introduce a concept of *cell states* which would better describe a cell depending on its context and function. However, an emerging challenge would be to connect *cell states* with historical *cell types*. [?].

Another aspect of cells, that I am not approaching in this thesis, is time. Cells are shaped by their environment, intrinsic and extrinsic events and can change states, functions etc. Can one cell belong to different cell types depending on its trajectory? How to include the dynamic aspect of the cells into the classification?

Thus, most scientists agree that used convention of cell types is not ideal and it is more matter of convenience than biological reality. This leaves a room to study cells and challenge existing classification. Describing cell types or cell states in the tumor microenvironment is extremely interesting as still little is known about the diversity of cell infiltrated in solid tissues.

1.1.2.4 Summary

Cancer is a disease concerning billions of people with a long history. Scientific community recognizes the role of the environment where the tumor cells find themselves as an important factor influencing tumor development, prognosis and response to treatment. TME is a complex environment that constantly interacts with tumor cells, where both tumor and TME influence and shape each other.

Over the years, many interactions are being discovered and cell types re-defined and described in their context. However, lots of mechanisms and interactions of TME remains unknown due to very heterogeneous nature of this microenvironment. This leaves room for a more extensive investigation of TME.

A therapeutic goal is target interactions that would be able to pivot the essential processes in tumorigenesis or tumor escape in order to put the cells “back on track” and facilitate anti-tumor therapies.

These goals can be met thanks to the improvement of investigation methods, data quality and abundance. I will discuss the most important data types used in this project to investigate the TME.

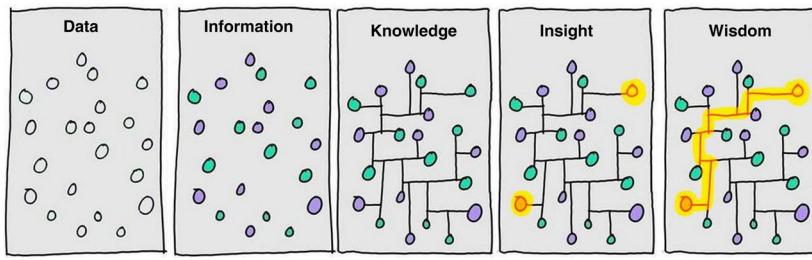


Figure 1.4: From Data to Wisdom. Illustration of different steps that it takes to go from *Data* to generating *Wisdom*. It highlights that generating data is not equal to understanding it and additional efforts are needed to generate value. Image authored by Clifford Stoll and Gary Schubert published by Portland Press Limited on behalf of the Biochemical Society and the Royal Society of Biology and distributed under the [Creative Commons Attribution License 4.0 \(CC-BY\)](#) in [?].

1.2 Quantifying and qualifying immune infiltration (data)

Nowadays, more and more biological data is produced. However, this proliferation of accessible resources is not proportional to generated insights and wisdom. In this thesis, I aim to generate *Knowledge* and *Insights* and we hope to generate some *Wisdom* (Fig. 1.4). In this section, we will introduce the foundation of our analysis: different data types that will be further discussed and explored in chapters that follow.

As discussed with the previous section cell-types, but also the whole systems can be described at different levels (along different axes). These different levels demand distinct technologies and strategies to be developed to enable the measurements. For instance, a phenotypic distinction between cells can be reached using FACS technology, for molecular profiles omic methods were developed and for ultrastructure microscopic methods. We need to approach biological systems from different angles as no one of these axis provide a complete picture of the studied system.

I will introduce most relevant data types that are used to study immune infiltration of tumors.

1.2.1 Cell sorting

1.2.1.1 Flow cytometry

Flow cytometry is a laser-based technology. It uses marker genes: cell surface proteins to sort cells in different compartments. Nowadays, it permits quantification of the abundance of up to 17 cell surface proteins using fluorescently labeled antibodies [?]. However this techniques is not free from bias, our knowledge about cell markers is limited and several markers may not be relevant in some context. Moreover, the scientific community did not clearly agree on the marker choice even for popular and well-studied cell types which introduced additional heterogeneity when independent studies are compared. Also, the quality of antibodies may influence the results of the FACS analysis. Besides those limitations FACS remains quite a popular method for

analyzing cells in complex tissues. It was among first methods that allowed molecular phenotyping of immune cells, a discovery of numerous subsets and their further functional interpretation.

1.2.1.2 Mass cytometry

Mass cytometry (also known as CyTOF) allows for the quantification of cellular protein levels by using isotopes. It allows to quantify up to 40 proteins per cell [?]. It also demands lower starting number of cells (1000 - 1000000), a realistic number that can be extracted from patient biopsy [?].

1.2.2 Microscope Staining

Using microscope technics, histopathological cuts are analyzed. The number of cells per a unit of area (i.e. mm²) is defined either manually by a human or through diverse image analysis algorithms.

Current pathology practice utilizes chromogenic immunohistochemistry (IHC) [?]. Multiplexed approaches allow identifying multiple markers in the same histopathology cut. Modern techniques like imaging mass cytometry using FFPE tissue samples uses fluorescence and mass cytometry to identify and quantify marker proteins [?].

The main advantage of aforementioned technics the number of cells that can be analyzed and the information about the spatial distribution of the different cell types. The limiting factor, as for cell sorting methods, is the number of markers (~10-100) and consequently a number of cell types that can be identified [?].

The cell sorting methods and microscope staining are usually considered as a gold standard for multidimensional data techniques. The reason why they are not applied at large scale is the cost but also quite laborious and time-consuming sample preparation demanding a fresh sample. In contrast, the omics methods propose a more scalable way to measure tumor microenvironment.

1.2.2.1 Tissue Microarrays

Tissue Microarrays aim to automatize “staining” techniques. A large number of small tissue segments can be organized in a single paraffin block where 100 tissue samples can be easily examined on one slide. A variety of molecular or microscopic method can be then applied to FFPE tissue including immunohistochemistry, FISH, and *in situ* hybridization [?]. It is a technique in between traditional imaging and omic high-throughput.

1.2.3 omics

In biological systems information is coded in the form of DNA that do not vary a lot between different individuals of the same species. To trigger a function in an organism, a part of the DNA is transcribed to RNA, depending on the intrinsic and extrinsic factors, and after additional modification messenger RNA (mRNA) is translated into a protein (i.e., digestive enzyme) that fulfill a role in the organism. The mRNA information (also called transcriptome) can be captured with experimental methods at high throughput (transcriptomics) and provides an approximation of the state of the studied system (i.e., a tissue). There is also information, not coded on the DNA sequence but in a pattern of chemical species that can regulate the state transition of DNA information. These additional regulators are called epigenome collectively and some of them, like methylation, can also be measured at high-throughput.

1.2.3.1 Transcriptome

Transcriptomics measures the number of counts of mRNA molecules using high-throughput techniques. mRNA is the part of the genetic information that should be translated into proteins. It reflects the activity of ongoing processes in a cell. In contrast to DNA, mRNA concentration can be highly variable [?]. This variability can be either “intrinsic” that reflect the stochastic process of cell machinery or “extrinsic” reflecting impact of factors upstream to mRNA synthesis [?].

Transcriptome can be measured by microarrays or RNA-seq NGS technology. Microarrays remain cost-efficient and popular technique designed in 90. There exist two and one fluorescent color probes, both representing different challenges in experimental design for batch effect removal. RNA-seq, in contrast, uses sequenced RNA to quantify the expression. As not only selected genes (probes) are quantified, it can be used to study unknown parts of the genome. RNA-seq is also characterized by lower background noise than microarrays.

Bulk transcriptome data are quite accessible nowadays. They can be obtained from either flash-frozen or formalin-fixed, paraffin-embedded (FFPE) tissue samples, including both surgically resected material and core needle biopsies [?].

The main flaw of transcriptomic data is that the reproducibility between different platforms is limited. As a result, direct comparison (direct merging, statistical difference tests) between two datasets produced by different platforms is not advised. There are 12 thousand genes that are matching between four sequencing platforms. Through gene names conversions much information is lost, and bias is introduced.

Different strategies can be adapted to analyze bulk transcriptome.

?] describes five groups of most popular approaches that can be applied to study transcriptome (Fig. 1.5). Despite a diversity of bioinformatic and statistical tools, the most popular differential approaches, mainly differential gene expression (DGE) based on the difference between two experimental conditions.

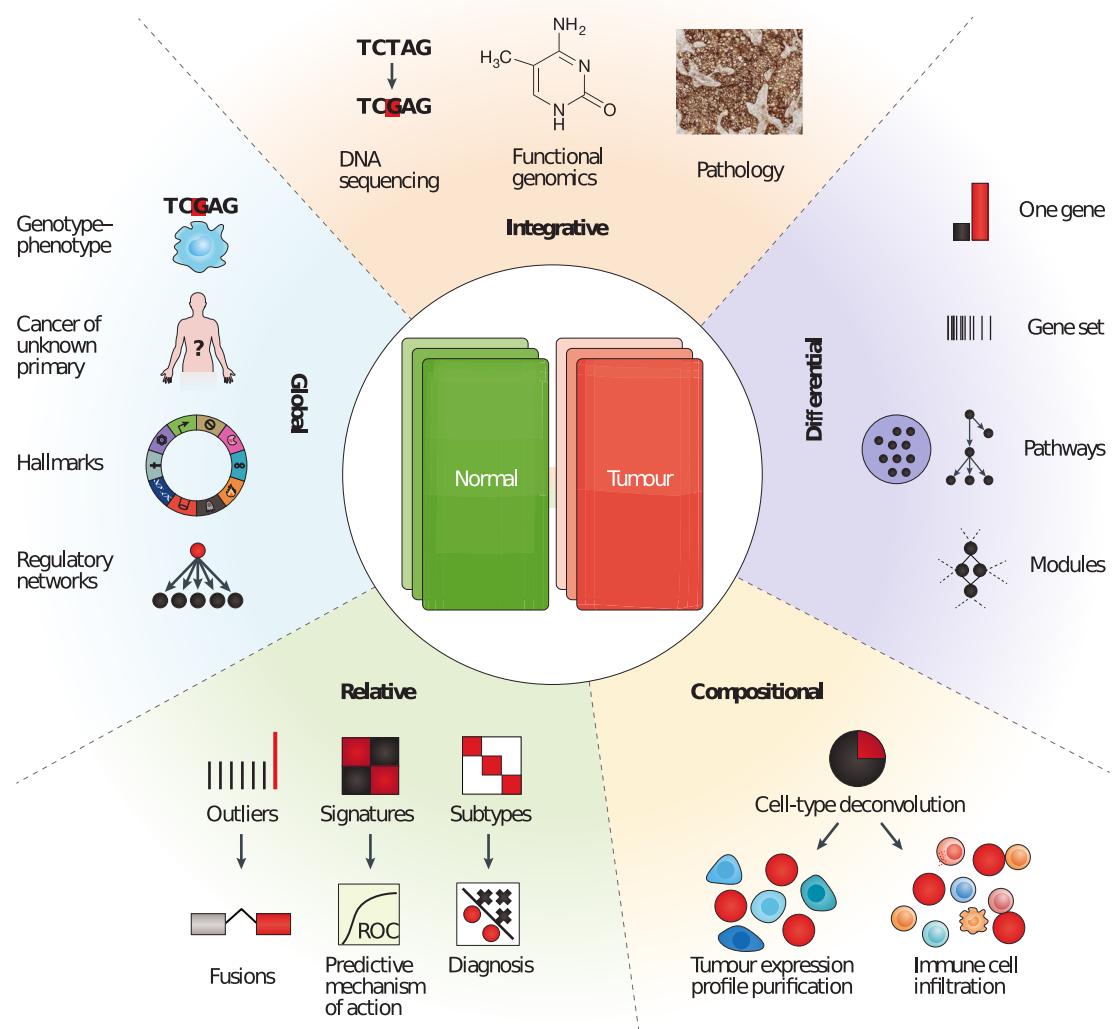


Figure 1.5: Five categories of RNA-seq data analysis. Differential analyses: comparing two (or more) conditions, Relative analyses: comparing to an internal reference (average, base level), Compositional analyses: inferring cell types or groups of cell types (i.e., tumor purity), Global analyses: pan-tissue and pan-cancer analyses and Integrative analyses: compiling heterogeneous data types. Reprinted by permission from Springer Nature [?] © 2018 Macmillian Publishers Limited, part of Springer Nature. All rights reserved.

RNA-seq data was proven to be a useful indicator for clinical applications [? ? ?]. Its utility for immune profiling was demonstrated in many studies through the use of transcriptomic signatures to predict immunotherapy response or survival [?].

In this work transcriptome data analysis falls into multiple categories: Compositional, Relative and aims to construct Global-level conclusions.

1.2.3.2 Single cell RNA-seq

Described above methods of process DNA from hundreds of thousands of cells simultaneously and report averaged gene expression of all cells. In contrast, scRNA-seq technology allows getting results for each cell individually. This is tremendous step forward enhancement of our understanding of cell heterogeneity and opens new avenues of research questions.

Continuous discovery of new immune subtypes has proven that cell surface markers that are used for phenotyping by techniques like FACS and immunohistochemistry cannot capture the full complexity. ScRNA-seq methods allow clustering known cell types in subpopulations based on their genetic features. ScRNA-seq is also able to capture particularly rare cell types as it requires much less of RNA material (1 ng isolated from 100-1000 cells) compared to ‘bulk’ RNA-seq (~ 1 µg of total mRNA transcripts). It also allows studying cells at high resolution capturing the phenotypes in much more refined scale than previously [?].

This new data type also brings into the field new challenges related to data processing due to the volume, distribution, noise, and biases. Experts highlight as the most “batch effect”, “noise” and “dropout effect” [?]. So far, there are no official standards that can be applied which makes data comparison and post-processing even more challenging. Up to date, there are around 70 reported tools and resources for single cell data processing [?]. A limited number of single-cell datasets of tumors are made publicly available, and more are to come.

One can ask why then developing computational deconvolution of bulk transcriptome if we can learn relevant information from single-cell data. Firstly, that single cell data do not provide a straightforward answer to the estimation of cell proportions. The coverage is not full and sequenced single cells are not entirely representative of the actual population. For instance, neutrophils are not found in scRNA-seq data because of they are “difficult to isolate, highly labile ex vivo and therefore difficult to preserve with current single-cell methods” [?]. Besides, a number of patients included in published studies of range <100 cannot be compared to thousand people cohorts sequenced with bulk transcriptome methods. This is mostly because single cell experiments are challenging to perform, especially in a clinical setting as fresh samples are needed [?]. Today, single cell technology brings very interesting “zoom in” perspective, but it would be incautious to make fundings from a restricted group of individuals universal to the whole population. Primary brake to the use of single cell technology more broadly might be as well the price that is nearly 10x higher for single cell sample compared to bulk [?].

In this work, we are using single cell data in two ways. Firstly, in Chapter 5 we compare immune cell profiles defined by scRNA-seq, blood and blind deconvolution (problem introduced in Im-

mune signatures section). Secondly, in Chapter 6 we use single call data of Metastatic melanoma generated by [?] to demonstrate heterogeneity of subpopulations of Macrophages and NK cells.

1.2.3.3 Epigenome

An epigenome can be defined as a record of the chemical changes to the **DNA and histone proteins** of an organism. Changes to the epigenome can provoke changes to the structure of chromatin and changes to the function of the genome [?]. Epigenome data usually contains information about methylation **CpG island changes**. In cancer, global genomic hypomethylation, CpG island promoter hypermethylation of tumor suppressor genes, an altered histone code for critical genes, a global loss of monoacetylated and trimethylated histone H4 were observed. Methylome profiles can also be used as a molecular signature of disease and potential diagnostic or predictive biomarker [?].

1.2.3.4 Copy number variation (CNV) and Copy number aberration (CNA)

The differences between human genome come in the majority from **Copy Number Variation** [?]. CNV regions constitute 4.8–9.7% of the whole human genome [?]. They can be reflected in structural variation that is duplication or deletion of DNA bases. CNV can affect a lot of base pairs of DNA code (deletion of more than 100 genes) and result in a phenotype change.

In addition, there can be distinguished, **Copy number alterations/aberrations (CNAs)** that are changes in copy number that have arisen in **somatic** tissue (for example, just in a tumor), in contrast to CNV that originated from changes in copy number in **germline** cells (and are thus in all cells of the organism) [?]. CNV and CNA profiles can be associated with diseases or cancer subtypes.

There exist disease-related exome panels that focus on regions with high copy variation, or the full exome can be sequenced using whole-exome sequencing (WES) [?].

1.2.3.5 Spatial transcriptomics

Spatial transcriptomics provides quantitative gene expression data and visualization of the distribution of mRNAs within tissue sections and enables novel types of bioinformatics analyses, valuable in research and diagnostics [?]

It combines RNA-seq technology with spatial labeling which allows having a bulk gene expression of 10-20 cells with given space coordinates within the sample. It allows to localize regions of highest gene expression and perform *Spatially Variable Genes* [?]). Some attempts were already made to combine Spatial Transcriptomics and scRNA-seq [?]. It remains an early-stage technique, and so far it is not widely used, but it might be a future of omics to add spatial information as it can be essential for many research problems.

1.3 From cancer phenotyping to immune therapies

This section outlines different methods of cancer immune phenotyping and progress in cancer therapies with a focus on immune therapies. It will link the ongoing research on TME with therapeutic potential.

1.3.1 Cancer immune phenotypes

Since 20s century physicians decided on common nomenclature that classifies tumors into distinct groups that are relatively homogenous or that share common characteristic important for treatment and prognosis. Tumor typing should help to predict prognosis better, to adopt a therapy to the clinical situation, to enable therapeutic studies which are essential in proving any therapeutic progress.

Most of the classifications are based on clinical data. Most common factors taken into account are the degree of local invasion, the degree of remote invasion, histological types of cancer with specific grading for each type of cancer, possibly various tumor markers, general status of the patient.

However, cancers with similar morphological and histopathological features reveal very distinct patterns of progression and response to therapy [?]. In the era of gene sequencing, gene and protein expression, as well as epigenome, can provide valuable complementary information. Therefore gene markers or proteomic abnormalities can be integrated into classification panel. One famous example is a gene signature *PAM50* [?] used for prediction of patients' prognosis in breast cancer, patented as a tumor profiling test.

Since the increase of importance of the immunotherapies, researches proposed several ways to classify tumors based on their microenvironment. Given different parameters describing TME, cancers can be sorted into groups that show similar characteristics. We will discuss most common frameworks that allow for phenotype cancers based on the TME.

The localization of the immune cells can be an indicator of the state and response to the therapy [?].

The most standard approach is to convey an analysis of histopathological cuts to asses the number of infiltrating lymphocytes (TILs). Two typical patterns are usually identified: "hot" - immune inflamed and "cold" - no active immune response [?].

[?] describe classification into inflamed and non-inflamed tumors, where non-inflamed phenotypes: can be further split into the immune-desert phenotype and the immune-excluded phenotype (Fig. 1.6). The inflamed phenotype is characterized by the abundant presence of immune cells: T cells, myeloid cells, monocytes in tumor margin. Along with the immune cells, due to their communication, a high expression of cytokines is characteristic for this phenotype. According to [?], this is a mark that an anti-tumor response was arrested by the tumor. The inflamed phenotype has shown to be most responsive to immunotherapies. In the immune-excluded phenotype,

the immune cells are present as well but located in the stroma [?], sometimes penetrating inside the tumor. However, when exposed to checkpoint immunotherapy, T cells do not gain the ability to infiltrate the tumor; therefore the treatment is inefficient. The immune-desert main features are little or no presence of immune cells, especially T cells. Surprisingly, these tumors have been proven to respond rarely to the checkpoint therapy [?]. In non-inflamed tumors, cytokines associated with immune suppression or tolerance are expressed.

A presence of immune phenotypes was confirmed by for example by ?] in colorectal cancer, where after deconvolution of bulk tumor profiles, patterns of immune and stromal cells abundance was matching four cancer subtypes. The good prognosis was related to cytotoxic response and bad prognosis to lymphocytes and cells of monocytic origin.

According to ?], the immunogenicity of the tumors can be explained by tumor-intrinsic factors and tumor-extrinsic factors. Tumor-intrinsic factors are the neoantigen load and frequency, the mutational load, the expression of immunoinhibitors and immunostimulators (e.i. PD-L1), and alteration of HLA class I molecules. Tumor-extrinsic factors include chemokines regulating T cell trafficking, infiltration of effector TILs and immunosuppressive TILs, and soluble immunomodulatory factors (cytokines).

1.3.2 Scoring the immune infiltration

Experimental techniques and computational tools enabled us to characterize and classify TME with multi-omics data. Here I present **a short list of most recent and influencing** analysis aiming to redefine tumor phenotypes based on the immune infiltration, with a focus on computational techniques.

1.3.2.1 Immunoscore

Jerôme Galon lab in Paris authors one of the most recognized scoring method, based on fluorescent images and names [Immunoscore](#). The Immunoscore ranges from 0 to 4 and it is based on the density of lymphocyte populations CD3/CD45RO, CD3/CD8, or CD8/CD45RO. It also takes into account the spacial position of the cells: the tumor core and margins [?]. It was successfully applied to colorectal cancer to predict patients' survival [?]. Since then, it resulted in numerous application to many cancer types. Immunoscore has been recently validated in a large cohort international independent study (14 centers in 13 countries) as a relevant prognostic score of time to recurrence, defined as the time from surgery to disease recurrence [?].

The immunoscore is an interesting indicator, especially in the scope of clinical applications, although it does not tell us a lot about underlying biology. It is also limited to a few cell types while it may be that in some cancer types or patients, the system requires more detailed or rich analysis of a larger panel of cells.

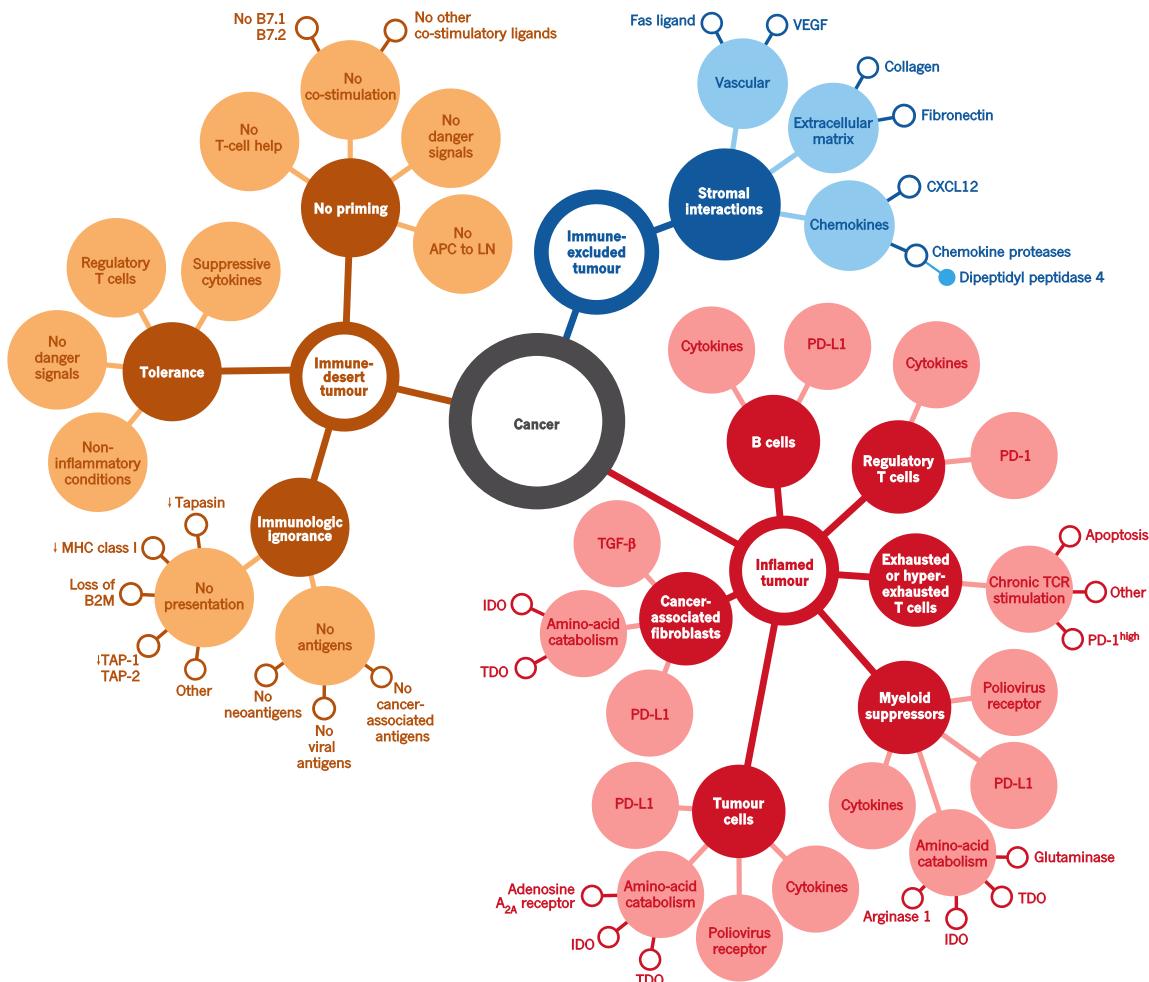


Figure 1.6: Cancer-immune phenotypes: the immune-desert phenotype (brown), the immune-excluded phenotype (blue) and the inflamed phenotype (red). The immune-desert phenotype is characterized by a paucity of immune cells and cytokines. In the immune-excluded phenotypes, the T cells are often present but trapped in the stroma, enabled to migrate to the tumor site. The immune-inflamed phenotype is rich in immune cells and the most responsive to the immune checkpoint therapies. Reprinted by permission from Springer Nature [?] © 2017 Macmillan Publishers Limited, part of Springer Nature. All rights reserved.

1.3.2.2 Spatiotemporal dynamics of Intratumoral Immune Cells of Colorectal Cancer

[?] published a quite complete, and supported with strong experimental evidence, immune landscape of colorectal cancer. Authors introduced *the immunome compendium* containing 577 cell-type-specific genes, derived from analysis of a significant corpus of publicly available data. They used it to analyze CRC large transcriptomic data (105 patients). Using qPCR (more sensitive technique than microarray) expression of 81 “representative” genes from the compendium was investigated in 153 CRC patients. This study validated correlation of markers of the same type and also revealed the correlation of different cell-type markers (i.e., T-cells and NK or Th and macrophages). The data matrix was grouped into 3 clusters which were corresponding to 1) tumor 2) adaptive 3) innate immune responses. Besides, spatial positioning of markers was visualized thanks to Tissue Microarray technology in samples from 107 CRC patients distinguishing marker densities in tumor center and tumor margin areas. This was followed by an in-depth study of chemokines expression and genomic alterations. Also, authors validated potential prognostic biomarkers in murine orthotopic CRC models.

In summary, using marker genes measured and visualized with different data types of CRC, a high inter-patient heterogeneity was observed. Adaptive immunity cells were associated with the core of the tumor and the innate ones with the tumor margin. A mechanism involving CXCL13, Tfh cells, B cells and IL-21 was identified as associated with good prognosis.

Authors suggest a dynamic dimension of the study which is in practice comparison between tumor stages. It can be argued that used time scale is too discrete and true dynamics cannot be reflected only along tumor stages in different patients. It is extremely challenging to access truly dynamic data for human tumor biopsies, but some efforts are made in the direction of inclusion of sequential biopsies [?] that allow better time resolution. In brief, the field is still waiting for the landscape of TME truly dynamic in space and time.

1.3.2.3 Immunophenoscore

Different approaches, sub-typing oriented, are based principally on gene expression patterns. Most commonly, machine learning supervised algorithms are trained to match known phenotype (established with microscopy or with clinical features) to genetic patterns, or an unsupervised clustering is used to discover new classification.

An example of well-formulated classification framework is Immunophenoscore [?], based on the publication of [?], where methylome, transcriptome and mutation of TCGA CRC dataset ($n = 598$) was used to describe *immunophenotypes*. Later on, it was reduced to gene expression indicator and summarised in the form of a score. This scoring scheme is based on the data of 20 solid tumors, using the expression of marker genes selected by a machine learning algorithm (random forest) for best prediction in each cancer. These indicators can be grouped into four categories:

- MHC molecules (MHC)

Table 1.1: Six immunological subtypes of cancer. The general characteristic of subtypes generated by [1] as described in the original publication.

Cluster	Features	Macrophage..lymphocyte	Th1.Th2	Proliferation	Intratumoral.heterogeneity	Other
C1	Wound healing	Balanced	Low	High	High	Highest M1 and CD8 T cells
C2	IFN- γ dominant	Lowest	Lowest	High	Highest	Highest Th17
C3	Inflammatory	Balanced	High	Low	Lowest	
C4	Lymphocyte depleted	High	Minimal Th	Moderate	Moderate	
C5	Immunologically quiet	Highest	Minimal Th	Low	Low	Highest M2
C6	TGF- β dominant	High	Balanced	Moderate	Moderate	Highest TGF- β signature

- Immunomodulators (CP)
- Effector cells (EC)
- Suppressor cells (SC)

The immunophenscore (IPS) is calculated on a 0-10 scale based on the expression of genes in each category. Stimulatory factors (cell types) impact the score positively and inhibitory factors (cell types) negatively. Z-scores \geq three were designated as IPS10 and z-scores ≤ 0 are designated as IPS0. A similar conceptual framework called *cancer immunogram* was proposed by [2]. It included seven parameters: tumor foreignness (Mutational load), general immune status (Lymphocyte count), immune cell infiltration (Intratumoral T cells), absence of checkpoints (PD-L1), absence of soluble inhibitors (IL-6, CRP), absence of inhibitory tumor metabolism (LDH, glucose utilisation), tumor sensitivity to immune effectors (MHC expression, IFN- γ sensitivity). [2] claim that the immunophenoscore can predict response to CTLA-4 and anti-PD-1.

Nonetheless, the details of the use of *cancer immunogram* in practice remain unclear and the result could be sensitive to patients' and data heterogeneity as no standardization was proposed. It should also be validated in a systematic, independent study.

1.3.2.4 The immune landscape of cancer

[1] performed a multi-omic analysis of TCGA datasets that allowed them to define six subtypes that are valid across cancer types (see Tab. 1.1).

Authors selected eight indicators to define these six phenotypes:

1. differences in macrophage or lymphocyte signatures
2. Th1:Th2 cell ratio
3. extent of intratumoral heterogeneity
4. aneuploidy
5. extent of neoantigen load
6. overall cell proliferation
7. expression of immunomodulatory genes
8. prognosis

These indicators were selected among many other indicators through machine learning (elastic net regression) for the best predictive power of survival.

All the data and computed parameters can be accessed at [CRI iAtlas Portal](#). Among the six phenotypes C3 (Inflammatory) has the best-associated prognosis while C1 (wound healing) and C2 (IFN- γ dominant), much less favorable outcome. This again illustrates the ambivalent nature of the immune system as the best, and the worst prognosis is associated with immunologically active tumors. C4 (lymphocyte depleted) and C6 (TGF- β dominant) subtypes had the worst prognosis. The content of immune cells was determined using different tools and data types (expression, DNA methylation, images, etc.) We can learn a lot from the study. However, it seems difficult to integrate the methods into an ordinary practice because different data levels are necessary for the same samples to compute all the indicators.

1.3.2.5 A pan-cancer landscape of immune-cancer interactions in solid tumors

A different classification was proposed by [?], also using TCGA data. They distinguished 17 immune infiltration patterns based on the immune cell proportions and 6 different clusters based on cytotoxicity measure across all cancer types (named immune-phenotypes) that were finally summarized in three groups: cytotoxic immune infiltrate, infiltrate with more immune-suppressive component and poor immune infiltrate. According to the analysis, one of the most critical factors is cytotoxicity. Tumors with high cytotoxicity were characterized by low clonal heterogeneity, with gene alterations regulating epigenetic, antigen presentation and cell-cell communication. The medium-level cytotoxic tumors had activated invasion and remodeling of adjacent tissue, probably favorable to immune-suppressive cells. The low cytotoxicity subgroup of tumors had altered: cell-cycle, hedgehog, β -catenin and TGF- β pathways. This result roughly overlaps with the one of [?]. The survival analysis based on the six immune-phenotypes revealed that for most cancer types, high cytotoxic tumors are associated with better survival. To evaluate tumor environment cells, authors used gene set variation analysis [?] with a set of pre-defined cell-type markers. Another important conclusion of [?] is that tissue of origin is not the only important factor shaping cell-type patterns in tumors. However, the least infiltrated tumors were lung, uterine and bladder cancers, while the most infiltrated were pancreatic, kidney, skin cancers and glioblastoma. They also analyzed cancer cell pathways after computational purification of tumor samples (subtraction of the immune signal) to better understand cancer signaling.

A different approach is to characterize tumors based on signaling pathways organized in functional modules.

1.3.2.6 Immune maps

Another way to summarize tumor phenotype can be through the use of molecular maps. [Atlas of Cancer Signaling Network \(ACSN\)](#) [? ?] is a pathway database that contains a collection of interconnected cancer-related signaling network maps. An additional feature is ACSN web-based

Google-maps-like visualization of the database. User data can be projected on the molecular map (for example gene/protein expression from user data can be paired with entities on the map.). ACSN 2.0 contains Cancer cell map and TME map (at the time: angiogenesis, innate immune map, T-cell signaling maps). All separate maps are available in [Navicell website](#). Through projection of the data on the innate immunity map, one can see if a tumor sample is characterized by pro- or anti-tumor activated pathways due to the organization of the map layout. Also, different CAF subtypes were characterized by the CAF specific map in [?]. Kondratova and colleagues (including myself) used the innate immune map to characterize NK and Macrophages subtypes (see Chapter Z).

1.3.2.7 Summary

Despite all scientific efforts, the gene expression-based classifications are not yet used in clinics. The measured multi-panel mRNA expression, which can be included into the category of In Vitro Diagnostic Multivariate Index Assay (IVDmia) [? ?], may be a future of TME-based cancer classification, diagnosis and treatment recommendation [?]. For this best tools need to be used to evaluate the state of TME and tumor-stroma-immune cells communication properly.

1.3.3 Immune signatures - biological perspective

A gene signature is

a single or combined group of genes in with a uniquely characteristic pattern of gene expression that occurs as a result of an altered or unaltered biological process or pathogenic medical condition [? ?].

They can be classified based on their form:

- metagene
- gene list
- weighted gene list

A term **metagene** or *eigen gene* describes an aggregated pattern of gene expression. The aggregation can correspond to simple mean of samples or can be obtained through matrix factorisation or source separation techniques, clustering. A metagene usually provides values for all measured genes (all probes) in contrast to a weighted gene list where weights are associated with selected genes.

Gene lists are simple enumeration of transcripts names or gene identifiers. Application of gene list is often limited to gene enrichment analysis tools or gene selection from the data.

An alternative is a **weighted gene list** or ranked gene list, where genes are ranked according to their importance. Often the ranks are obtained through comparison between two conditions

or test/control. They can be also based on absolute gene expression values [?]. One possible problem with this weighted gene list can be platform dependence.

There exist a big choice of databases storing collections of signatures. They contain gene expression and other genomic data such as genotype, DNA methylation, and protein expression data attributed to some condition of reference. A big collection of immune signatures are regrouped by [Immunological Genome Project \(IGP, ImmGen\)](#) [?]. Gene expression of protein coding genes measure in mice immune cells, ex vivo, in different conditions (drug treatment, perturbations) were regrouped in this ressource. A different ressource [Immuno-navigator](#) [?] that stores information about human and murine immune genes and co-expression networks. [ImmuneSigDB](#) is a collection of gene-sets that describe immunity and inflammation in transcriptomic data [?] and a part of popular MSigDB ressource used commonly for gene set enrichment analysis (GSEA) [?].

They can also be classified based on their use:

- prognostic signatures
- predictive signatures
- diagnostic signatures
- specific signatures

The *prognostic* signatures can distinguish between patients with a good or from patients with bad prognosis when deciding to assign a patient to a therapy.

The *predictive* signatures are able to predict treatment benefit between experimental and/or nontraditional treatment groups vs. control, i.e. in clinical trials [?].

The *diagnostic* signature, also called *biomarkers* can be used for detection of a disease in a patient, like for example in blood tests.

The *specific* signatures should describe with robustness and reproducibility the same group of cells, or patients, or condition with respect to other considered groups. For instance, in the context of cell-types, among studied cell-types a specific signature will distinguish only one cell type. In the context of cancer subtypes, it will indicate clearly one subtype among others.

Examples of predictive and prognostic gene signatures, used in clinical practice are Oncotype DX, EndoPredict, PAM50, and Breast Cancer Index for breast cancer [?].

Studies discussed in this Chapter showed plausible importance of immune-related signals in cancer therapy. However, there is no immune-related gene signatures used in clinical practice currently. This can be because of the lack of consistency of genes, both within the same tumor type and among different tumors that can be found in the signatures [?]. Difference in gene expression of different cell populations were found even intra- and interlabs. This difference can be due to confounding factors like stress or to contamination [?].

In many studies *specific* signatures of cell types are used. They seem to be good in discriminating between broad lineages of cell type, such as lymphoid and myeloid. Although thier capacity to

describe cell states and cell subtypes is more discutable [?]. Another matter is that cell type signatures are often obtained in model organisms or extracted from different tissue (i.e. blood-derived signatures vs cancer-derived signatures).

the gene expression profiles of tumour-associated immune cells differ considerably from those of blood derived immune cells [?]

With emergence of single-cell signatures, there are new horizons of gene signatures to be discovered. Especially signatures of rare cell types in solid tissues. Yet, it is up to researches to cross validate single cell signatures with different types of data as scRNA-seq is not free of platform and post-processing bias.

Immune signatures will be also discussed as a part of deconvolution pipeline in the Chapter 2 under the section about *basis matrix* in mathematical terms.

1.3.4 Cancer therapies

Cancer is a complex disease. Up to date, no uniform and fully effective treatment were proposed, and usually different strategies are tested to kill tumor cells. **Surgery** is one of the oldest methods. The cancer is removed from the patient body. There are different ways, more or less invasive, that it can be performed. It is usually applied for solid tumor contained in a small area. **Radiation Therapy** uses high doses of radiation to eliminate tumor cells and shrink tumor mass. It can be applied externally or internally. **Chemotherapy** uses a drug (or a combination of drugs) that kill cancer cells, usually altering cell proliferation and growth. The drawback of radiotherapy and chemotherapy are substantial side effects. **Hormone therapy** modulate hormone levels in the body in order to inhibit tumor growth in breast and prostate cancers. In leukemia and lymphoma, can be applied **stem cell transplants** that restore blood-forming stem cells destroyed by the very high doses of chemotherapy or radiation therapy that are used to treat certain cancers.

Alternatively, **targeted therapies** represent a more focused strategy that aims to be more efficient and cause fewer side effects than systematic therapies. Two main types of targeted therapies are small-molecule drugs and monoclonal antibodies. Targeted therapies usually aim to stimulate/inhibit a selected molecular function. Particular types of targeted therapies are **Immunotherapies**. Through activation/inhibition of immune regulatory pathways, it stimulates the immune system to destroy malignant cells. A continuation of targeted therapies is **precision medicine approach**. It is based on genetic information to specify patient's profile and find a suitable treatment. A number of innovative treatments targeting a specific change in tumor ecosystem are being tested presently in precision medicine clinical trials [?].

1.3.5 Recent progress in immuno-therapies

The immunotherapies, in contrast with other types of cancer therapies discussed in the previous section, aim to trigger or restart the immune system to defend the organism and attack the

malignant cells without provoking persisting inflammation state [?]

The idea of stimulating the immune system to fight malignant cell was not born recently. For a long time, a possibility of development of an anti-cancer vaccine has been investigated. Unfortunately, this idea faced two essential limitations 1) lack of knowledge of antigens that should be used in a vaccine to stimulate cytotoxic T cells successfully) the ability of cancer to block the immune response also called *immunostat*. Despite those impediments works on anti-tumor vaccines do not cease [?]. A very recent promising an in-situ anti-tumor vaccine was proposed by Sagiv-Barfi et al. [?]. The therapy tested in mice would be based on local injections of the combination of “unmethylated CG-enriched oligodeoxynucleotide (CpG) - a Toll-like receptor 9 (TLR9) ligand and anti-OX40 antibody. Low doses of CpG injected into a tumor induce the expression of OX40 on CD4+ T cells in the microenvironment in mouse or human tumors. An agonistic anti-OX40 antibody can then trigger a T cell immune response, which is specific to the antigens of the injected tumor”. Sagiv-Barfi et al. claim this therapy could be applied to all tumor types, as long as they are leucocyte-infiltrated. As a local therapy, in situ vaccination should have fewer side-effects than systematic administration. It is now undergoing clinical trials to test its efficiency in human patients.

Another idea involving using the immune system as a weapon to fight cancer would be the use of genetically modified patient's T-cells, carrying CARs (chimeric antigen receptors) [?]. After an extended period of small unsuccessful trials, recently in 2017, two CAR T-cell therapies were accepted, one to “treat adults with certain type of large B-cell lymphoma” [?], other to treat “children with acute lymphoblastic leukemia (ALL)” [?], which are, at the same time, the first two gene therapies accepted by FDA.

However, the two most promising immuno-related strategies with proven clinical efficiency are based on blocking so-called immune checkpoint inhibitors: cytotoxic T-lymphocyte protein 4 (CTLA4) and programmed cell death protein 1 (PD-1). The anti-CTLA4 antibodies blocks repressive action of CLTA4 on T-cells and they become therefore activated. It was shown efficient in melanoma patients and accepted by FDA in 2015 as adjuvant therapy for stage III metastatic melanoma patients [?]. PD-1 is a cell surface receptor of T cells, that binds to PD-L1/PD-L2. After binding, an immunosuppressive pathway is activated and T cells activity is dampened. An action of an anti-PD-L1 antibody is to prevent this immune exhaustion [?]. A stepping stone for anti-PD-L1 therapies was approval of Tecentriq (atezolizumab) for Bladder cancer [?] and anti-PD1 Keytruda (pembrolizumab) initially accepted for NSCLC and further extended to head and neck cancer, Hodgkin's lymphoma, gastric cancer and microsatellite instability-high cancer [?]. Since that breakthrough, other anti-PD-L1 or anti-PD1 antibodies were accepted or entered advanced stages of clinical trials [?]. A short history of immunotherapy FDA-accepted treatments can be found in Fig. 1.7

The main drawback of immunotherapies is heterogeneity of response rate, which can vary, i.e., from 10–40% in case of PD-L1blocking [?], suggesting that some patients can have more chances than others to respond to immune therapy. So far, it has been shown that anti PD-L1 therapies work more effectively in T cell infiltrated tumors with the exclusion of Tregs because of lack of difference in expression of FOXP3 in responding and the non-responding group of patients [?]

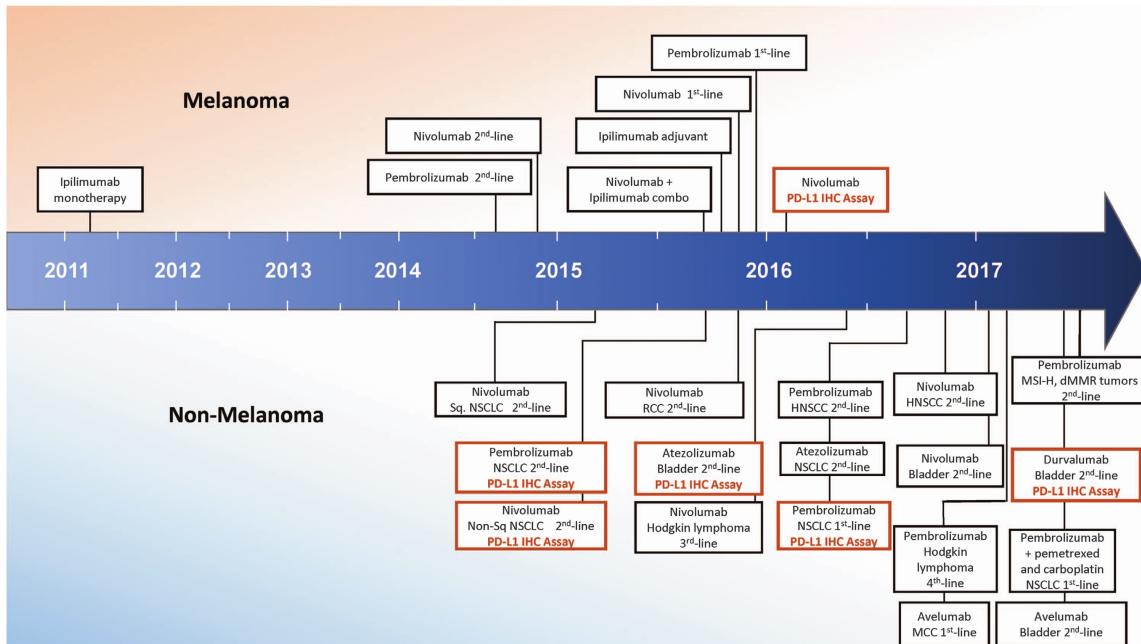


Figure 1.7: This timeline describes short history of FDA approval of checkpoint blocking immunotherapies up to 2017. Reprinted by permission from Springer Nature [?] Macmillan Publishers Limited, part of Springer Nature. All Rights Reserved.

[]. Also, some light has been shade by [?] who connected mutational rate of cancer cells to the chances of response to immunotherapy.

Despite those findings, the precise qualifications of patients that should be sensitive to immunotherapy are not defined [?]. As most patients do not answer to immunotherapies, it stimulates researches to look for better biomarkers and patient stratifications, and pharmaceutical industries to discover new immune checkpoints based therapies.

1.4 Summary of the chapter

Cancer remains a critical health problem of our era that touches many people. Tumor cells are interacting with their microenvironment (called Tumor Microenvironment (TME)) including normal cell, stromal cells and a variety of immune cells. These cells can have a role in disease progression and response to treatment. A modern approach to modulate TME was proposed through an application of immune therapies.

A new way to classify cancers based on their TME is called immunophenotyping. Widely used TCGA data contains many different data types, but not all of possible data types. Before entering a clinical practice, the immunophenotyping approaches will need to face important challenges. Would addition of a new data type (i.e., FACS) change the patients classification? Can the simi-

lar results be found in non-American patient cohorts? What if different technologies are used, if data are not normalized uniformly, would it change the conclusions? What if not all data types available in TCGA are not produced for other patients? How these classification can be reproduced for smaller cohorts? Can these complex classification schemes be reduced to a few easy measurable indices? It is important to acknowledge the authors for their remarkable work. However it is also crucial to remember we are biased by the piece of the truth (type of data) we use, that can be on the top biased with technical and experimental design.

To produce a very detailed system-level view of the TME with traditional experimental techniques an uncountable amount of work and resources would be necessary. Using omic techniques system approach is possible to reduce the time and resources. However, to embrace fully the data complexity, computational tools are indispensable. From the data generation to the analysis, different statistical and mathematical challenges need to be faced before arriving at valid biological results and interpretations.

As I will present in the next chapter, in order to solve the problem of extraction of cell-type heterogeneity from cancer bulk omic data, a number of approaches were developed.

Chapter 2

Mathematical foundation of cell-type deconvolution of biological data

In the previous chapter, I presented state-of-art of the current immuno-oncology research that has to embrace the vast complexity of cancer disease and the immune system. One part of this complexity can be explained by the presence and quantities of tumor-infiltrating immune cells, their interactions with each other and the tumor.

In this chapter, I will discuss how mathematical models can be used to extract information about different cell-types from 'bulk' omics data or how to de-mix mixed sources composing the bulk samples. To start with, I will introduce you to basic concepts of machine learning. Then I will focus on approaches adapted for cell-type deconvolution. In a literature overview, I will depict the evolution of the field as well as discuss the particularities of different tools for estimating presence and proportion of immune cells within cancer bulk omic data.

2.1 Introduction to supervised and unsupervised learning

Machine learning (ML) is a field of computer science where a system can learn and improve given an objective function and the data.

Mitchell gave a popular definition of machine learning in 1997:

Machine learning: A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E.

— Mitchell in 1997 [?]

Term *Artificial intelligence* (AI) is often used by the media or the general public to describe machine learning. Indeed ML can be considered as a branch of AI, together with computer vision

and deep neural networks applications. However, commonly ML and AI are used interchangeably by the broad public.

ML is applied commonly in many fields of science and industry. I will not discuss here subtle differences between machine learning, statistical learning, computational statistics and mathematical optimization.

In general, algorithms can be divided into groups given the application:

- classification - aims to assign observations to a group (discrete variable)
- regression - aims to predict a continuous response of an input (continuous variable)
- clustering - aims to divide data into groups that are related to each other based on a distance

Another critical distinction can be made given the inputs to the algorithm. Here, I present the differences between supervised and unsupervised learning.

2.1.1 Supervised learning

Supervised learning can be described as “the analysis of data via a focused structure” [?]. The primary task is to predict an output given the inputs. In the statistical language, the inputs are often called the predictors or the independent variables. In the pattern recognition literature, the term features are preferred. The outputs are called the responses, or the dependent variables. [?]

The initial data is divided into two sets: training and test. First, the model is trained with correct answers on the training data (learning to minimize the error), and then its performance is evaluated on the test data.

Among widely used classifiers there are Support Vector Machines (SVM), partition trees (and their extension random forests), and neural networks. For regression, it is common to encounter linear regression, boosted trees regression,

2.1.2 Unsupervised learning

In Unsupervised learning is given the data and is asked to segment the data given a particular constraint. However, the true segments of the data are not known. Therefore an unsupervised algorithm aims to unveil the “hidden structure” of the data or latent variables.

One group of unsupervised learning are descriptive statistic methods, such as principal components, multidimensional scaling, self-organizing maps, and principal curves. These methods aim to represent the data most adequately in low-dimensional space [?].

Another group is clustering algorithms. Clustering is the way to create groups (multiple convex regions) based on the intrinsic architecture of the data. These groups are not necessarily known beforehand but can be validated with the domain knowledge. Popular clustering algorithms are k-means, hierarchical clustering, mixture density-based clustering [?].

In both descriptive statistics and clustering, one important parameter (i.e. denoted k) is the number (number of factors, variables, clusters) to which the data should be decomposed. Different algorithms and applications can propose an automatic choice of k based on formal indexes or previous knowledge, in others, the user needs to provide the k .

2.1.3 Low-dimensional embedding for visualization

There is a common confusion, often seen in computational biology, between dimension reduction and clustering. This confusion is highly pronounced with, a popular in biology, algorithm: T-distributed Stochastic Neighbor Embedding (t-SNE) [?]. t-SNE works in 2 main steps: (1) a probability distribution over pairs of high-dimensional objects is computed in such a way that similar objects have a high probability of being picked, whilst dissimilar points have an extremely small probability of being picked, (2) t-SNE defines a similar probability distribution over the points in the low-dimensional map, and it minimizes the Kullback–Leibler divergence between the two distributions with respect to the locations of the points in the map. It is not reliable to use t-SNE for clustering as it does not preserve distances. It can also easily overfit the data and uncover ‘fake’ or ‘forced’ patterns. Therefore, a clustering should not be applied to t-sne reduced data. An alternative to the t-SNE method is recently published Uniform Manifold Approximation and Projection for Dimension Reduction (UMAP) [?] - that is based on Laplacian eigenmaps, highly scalable, reproducible and recently applied to biological data [?]. Older used alternatives are ISOMAPS (non-linear dimension reduction) or PCA (Principal components analysis). For any non-linear dimension reduction method, it is not recommended to use clustering *a posteriori*. Clusters should be computed on original data, and then the cluster labels can be visualized in low-dimensional embedding.

2.2 Types of deconvolution

One specific application of mathematical/statistical tools is deconvolution of mixed signals.

According to a mathematical definition:

Deconvolution : *the resolution of a convolution function into the functions from which it was formed in order to separate their effects.* [?]

Alternatively, in plain English:

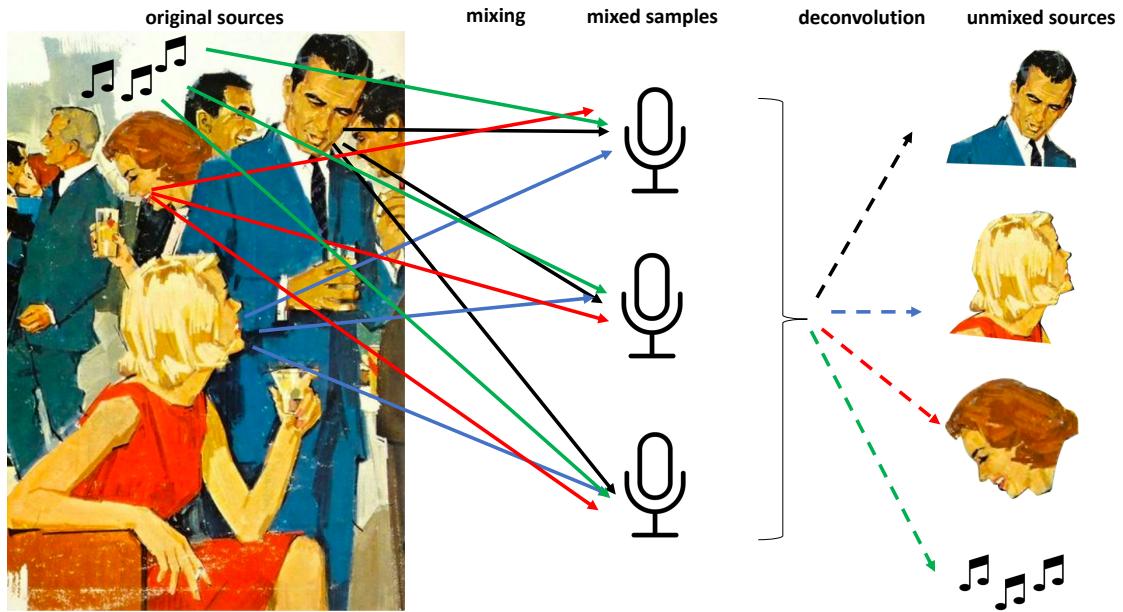


Figure 2.1: Illustration of the cocktail party problem. During a cocktail party voices of participants can be recorded with a set of microphones and then recovered through blind source separation. The illustration purposes only four sources are mixed with three microphones, in reality, the analysis can be performed with many sources. However, a number of samples (microphones) should be higher than the number of sources (contrary to the illustration).

a process of resolving something into its constituent elements or removing complication [?]

The similar problem of mixed sources can be encountered in other fields, i.e., signal processing, also known under the name of “**cocktail party problem**.” In the cocktail party problem, at a party with many people and music, sound is recorded with several microphones. Through blind source separation, it is possible to separate the voices of different people and the musical background (Fig. 2.1) [?].

The same concept can be transposed to the bulk omic data, each biological species (like gene) is a cocktail party where each sample is a microphone that gathers mixed signals of different nature. The signals that form the mixtures can be different depending on the data type, and the scientific question asked.

In general, the total bulk data can be affected by three abundance components [?]:

1. sample characteristic (disease, clinical features)
2. individual variation, genotype-specific or technical variation
3. presence and abundance of different cell types expressing a set of characteristic genes

Many scientists invested their efforts in order to dissect the bulk omic data into interpretable biological components.

In scientific literature, there can be encountered three main understanding of tumor deconvolution:

- **estimating clonality:** using genomic data is it possible to trace tumor phylogeny raised from mutations and aberrations in tumor cells; therefore it is dissecting *intra-tumor* heterogeneity (i.e., using transcriptomic data [?], or more often CNA data (see Section 2.4.2)
- **estimating purity:** deconvolution into the tumor and immune/stroma compartments, often aiming to “remove” not-tumor signal from the expression data, can be performed with different data types, the most reliable estimations are usually obtained from CNA data (see Section 2.4)
- **estimating cell-type** proportions and/or profiles from bulk omics data, most of works were performed on transcriptome data (see Section 2.3) and some on the methylome data (see Section 2.4.1)

These three types of deconvolution can be performed on the bulk omics data. Here we will focus on cell-type deconvolution models using bulk transcriptome. I will also briefly introduce deconvolution models applied to other data types (methylome and CNA).

2.3 Cell-type deconvolution of bulk transcriptomes

The idea of un-mixing the bulk omic profiles is documented to first appear in an article of [?] as a way to

infer the gene expression profile of the various cell types (...) directly from the measurements taken on the whole sample

In the primary hypothesis [?], a mixture of signals from TME in transcriptomic samples can be described as a linear mixture.

$$X = SA \quad (2.1)$$

Where in Equation (2.1) X is microarray data matrix of one biological sample, A are mixing proportions, and S is the matrix of expression of genes in each cell type.

Algebraically the same problem can be formalized as the latent variable model:

$$\begin{aligned} \forall i \in \{1, M\}, \forall j \in \{1, N\} \\ x_{ij} = \sum_{k=1}^K a_{kj} * s_{ik} + e_{ij} \end{aligned} \quad (2.2)$$

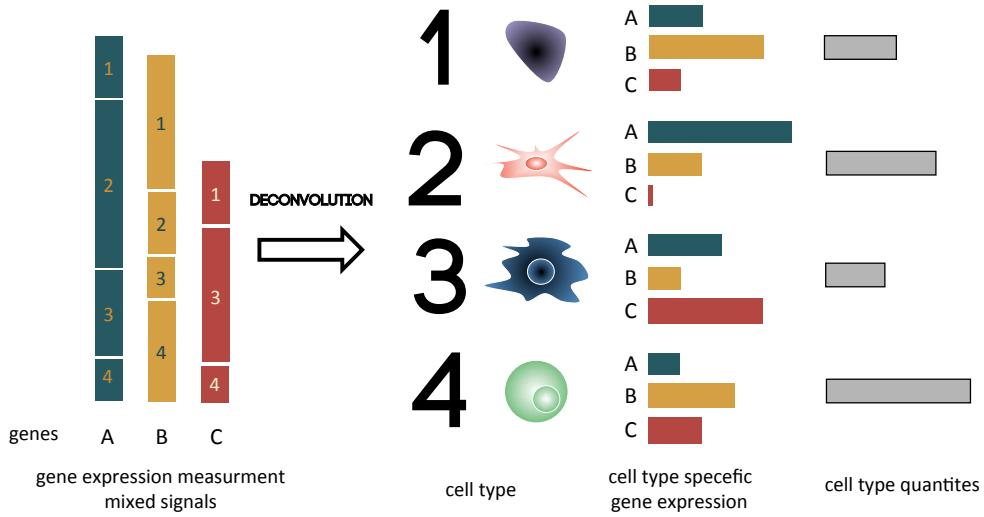


Figure 2.2: Principle of the deconvolution applied to transcriptome Graphical illustration of the deconvolution of mixed samples. Starting from the left, gene expression of genes A B C is a sum of expression of cell types 1, 2, 3, 4. After deconvolution, cell types are separated, and gene expression of each cell type is estimated taking into account cell type proportions.

Where x_{ij} is expression of gene i in sample j , a_{kj} is the proportion of cell type k in sample j and s_{ik} is the expression of the gene i in the cell type k , K total number of cell types, N total number of samples, M total number of genes. The error term e_{ij} cannot be directly measured.

The goal of deconvolution is to reverse these equations and starting from the mixture infer the A (or a_{kj}) and S (or s_{ik}).

Graphically the deconvolution of bulk gene expression can be depicted as in Fig. 2.2.

However, in this model, either the mixing proportions, number of mixing sources or an array of specific genes need to be known. While, in the real-life case, only X is truly known. Therefore, developed models proposed various manners for estimating the number of mixing sources and their proportions, or the specific cell type expression.

Why there is a need for cell-type deconvolution approaches?

- for differential gene expression analysis, to avoid confusion between a studied condition (i.e. disease impact) and cell-type abundance (change in gene expression due to the change of cell proportions)

- difference in gene expression in one cell type can be blurred by the presence of other cells expressing the gene
- to obtain information about a fraction of given component in the sample
- to study potential interactions between cell types in the studied context
- to infer context-specific profile or signature

2.3.1 Literature overview

In order to answer general and specific need for cell-type deconvolution of bulk transcriptomes researches produced a large collection of tools. I have collected all (to my knowledge) articles published in journals or as a pre-print (up to May 2018) that propose original models/tools of **cell-type deconvolution of bulk transcriptomes** (Tab. 2.1). Therefore clonal deconvolution methods are not included in this overview. The transcriptome-based purity estimation methods are included as many of them proposed an initial 2-sources model that could be, at least in theory, extended to multiple sources model. Also, I did not include cell-type deconvolution methods of other data types (such as methylome). A separate section 2.4 is dedicated to non-transcriptome methods.

####Growth of the field

The Table 2.1 contains 64 (including mine) deconvolution methods. It can be observed (Fig. 2.3) that since the beginning of my thesis (2015) the number of publications has doubled (64 publications in 2018 vs. 33 in 2014). Also, from 2014 on, more methods are published every year. In Fig. 2.3 *hallmark* publications are indicated in red above their year of publication. The three most popular methods (based on number of citations/number of years since publication) are CIBERSORT [?] (2015, total number of citations: 343 and 88.75 citations per year), ESTIMATE [?] (2013, total number of citations: 266 and 44.33 citations per year), and csSAM [?] (2010, total number of citations: 286 and 31.77 citations per year). It can be noticed that the high impact of the journal plays a role, the top 3 cited methods were published in *Nature Methods* and *Nature Communications* followed by Virtual Microdissection method [?] (2015) published in *Nature Genetics*. However, the fifth most cited publication [?] (2009, a total of 207 citations) appeared in *PLOS ONE*. As the index is a bit penalizing for recent publications, among commonly cited tools after 2015 are MCPcounter with 42 citations (2016, 32 without self-citations) and xCell with 14 citations (2017, 11 without self-citations). A big number of publications with a low number of citations were published in *Oxford Bioinformatics* or *BMC Bioinformatics* which underlines the importance of publishing a computational tool along with an important biological message rather than in a technical journal in order to increase a chance to be used by other researchers.

2.3.1.1 Availability

Another essential aspect is the availability of the tool. One-third (in total 21) methods do not provide source code or a user-interface tool to reproduce their results. Among those articles, 13 was published before 2015. Therefore, it can be concluded that the pressure of publishers and research

community on reproducibility and accessibility of bioinformatic tools gives positive results. ?], authors of semi-supervised NMF method [?], published *CellMix: a comprehensive toolbox for gene expression deconvolution* where he implements most of previously published tools in R language and group them in the same R-package. This work tremendously increased the usability of previously published deconvolution methods. The CellMix package is one of the state-of-the-art work on deconvolution that regroups algorithms, signatures and benchmark datasets up to 2013.

2.3. CELL-TYPE DECONVOLUTION OF BULK TRANSCRIPTOMES

67

Table 2.1: Summary of methods for cell-type deconvolution of bulk transcriptome. Data gathered based on PubMed and google scholar search in May 2018.

name	data	type	doi	year	application	availability	out_profiles	out_proportions	category	language	citations	pop_index	previously/covered
CSV4 scores	RNA-seq	unsupervised	https://doi.org/10.158709/042z/cb13509	2016	Cancer transcriptome	NA	FALSE	environment	unknown	0	100	FALSE	
M4cont	MA	supervised	https://doi.org/10.18646/3894.008	2018	Blood	https://hextoolshed.g2bix.psu.edu/repository/repository_id/4ef9a19ab263e57ba8dchangeset_revision@v309w02a	TRUE	regression	R, web tool	0	0.00	FALSE	
ADVOCATE	RNA-seq	supervised	https://doi.org/10.1007/s00979-018-1120-6	2018	Cancer transcriptome	NA	TRUE	probabilistic	R	0	0.00	FALSE	
DTS	scRNA-seq	supervised	https://arxiv.org/abs/1801.08474v1	2018	Cancer transcriptome	https://github.com/CancerGenomics/CellDeconv	TRUE	regression	unknown	0	0.00	FALSE	
Celltypepusher	MA + DNA-seq	unsupervised	https://doi.org/10.1101/232067	2018	yeast cell cycle	https://github.com/CancerGenomics/CellDeconv	TRUE	convolutional	unknown	0	0.00	FALSE	
dtangle	MA + RNA-seq	supervised	https://doi.org/10.1001/29082	2018	Blood	https://github.com/CancerGenomics/CellDeconv	FALSE	regression	R	0	0.00	FALSE	
DecoNC4	MA + RNA-seq	unsupervised	https://doi.org/10.2389/vodo.125069	2018	Cancer transcriptome	https://urlzuracewinski.github.io/DecoNC4	TRUE	TRUE	matrix factorisation	R, matlab	0	0.00	FALSE
xCell	MA + RNA-seq	supervised	https://doi.org/10.1101/180909-017-1546-1	2017	Cancer transcriptome	https://urlzuracewinski.github.io/DecoNC4	TRUE	enrichment	R, web tool	15	750	FALSE	
BioQ-Ch	MA + RNA-seq	supervised	https://doi.org/10.7554/bioRxiv.26476	2017	Cancer transcriptome	https://www.bionano.org/jacobson/buchheit/BioQChml	FALSE	TRUE	enrichment	R, web tool	6	330	TRUE
EPIC	RNA-seq	supervised	https://doi.org/10.20288/vodo.299	2017	Cancer transcriptome	https://github.com/Cellexy/Epic/	FALSE	TRUE	regression	R	4	200	FALSE
Estimation of immune cell content	scRNA-seq	supervised	https://doi.org/10.1038/s41467-017-02289-3	2017	Cancer transcriptome	NA	FALSE	regression	unknown	3	150	FALSE	
Enumerate�blood	MA	supervised	https://doi.org/10.1101/232064-016-3460-1	2018	Blood gene expression	https://github.com/enumrate/enumrateblood	TRUE	probabilistic	R	2	100	TRUE	
Immunologos	MA	supervised	https://doi.org/10.1101/232065	2018	Blood gene expression	https://github.com/enumrate/enumrateblood	FALSE	regression	R	1	0.00	FALSE	
quantSeq	RNA-seq + Images	supervised	https://doi.org/10.1007/s00979-017-2349-6	2017	Cancer transcriptome	https://doi.org/10.1007/s00979-017-2349-6	FALSE	regression	web tool	1	0.50	FALSE	
SMC	MA	unsupervised	https://doi.org/10.1371/journal.pone.019867	2017	Tissue mixtures	https://github.com/maynardmcgregor/SMC	TRUE	probabilistic	matlab	1	0.50	FALSE	
Modular dissection index	MA + RNA-seq	supervised	https://doi.org/10.1101/232066	2017	Skin tumour	https://github.com/MDKurnat/MDiscoIndex	TRUE	TRUE	enrichment	R	1	0.00	FALSE
Demix	MA + RNA-seq	supervised	https://doi.org/10.1101/232065	2017	Cancer transcriptome	https://github.com/MDKurnat/MDiscoIndex	TRUE	probabilistic	matlab	0	0.00	FALSE	
Post-modified non-negative matrix factorization	RNA-seq	unsupervised	https://doi.org/10.20288/vodo.299	2017	Cancer transcriptome	NA	TRUE	matrix factorisation	matlab	0	0.00	FALSE	
infra	RNA-seq	supervised	https://doi.org/10.1101/232067	2018	Cancer transcriptome	https://github.com/hammertech/Infra	TRUE	probabilistic	Stan	0	0.00	FALSE	
MCDCoupler	MA	supervised	https://doi.org/10.1101/232068	2018	Cancer transcriptome	https://github.com/hammertech/MDCCoupler	TRUE	environment	R	42	400	TRUE	
scSEA applied to oral cell carcinoma	RNA-seq	unsupervised	https://doi.org/10.1101/232069	2018	Cancer transcriptome	NA	FALSE	regression	unknown	4	100	TRUE	
CAM	MA	unsupervised	https://doi.org/10.1101/232070	2018	Cancer transcriptome	https://doi.org/10.1101/232070	TRUE	convex hull	R, java	12	4,00	TRUE	
Immune Quant	undefined	supervised	https://doi.org/10.1101/232071	2018	Human tissues	https://doi.org/10.1101/232071	TRUE	regression	web tool	5	167	TRUE	
VOCAL	MA + QWAS	supervised	https://doi.org/10.1101/232072	2018	Uterus tissue	https://doi.org/10.1101/232072	FALSE	regression	R	5	167	TRUE	
CellMapper	MA	semi-supervised	https://doi.org/10.1101/232073	2018	Brain tissue	https://doi.org/10.1101/232073	TRUE	matrix factorisation	R	5	167	TRUE	
contamINE	RNA-seq	supervised	https://doi.org/10.1101/232074	2018	Tumor purity	https://github.com/hammertech/contamINE	TRUE	probabilistic	R	4	133	TRUE	
IM3D	RNA-seq	supervised	https://doi.org/10.1101/232075	2018	Cancer transcriptome	https://github.com/hammertech/IM3D	TRUE	enrichment	unknown	0	0.00	FALSE	
CBERS-2DT	RNA-seq	supervised	https://doi.org/10.1101/232076	2018	Cancer transcriptome	https://github.com/hammertech/CBERS-2DT	TRUE	probabilistic	web tool	86	850	TRUE	
Virtual Microdissection	MA	unsupervised	https://doi.org/10.1101/232077	2018	detection of cancer and storms in POAC (TCGA)	https://doi.org/10.1101/232077	TRUE	matrix factorisation	R, C++, Fortran	28	7,00	TRUE	
CellCODE	MA	semi-supervised	https://doi.org/10.1101/232078	2018	Blood diseased tissues	https://www.pitt.edu/~mhilka/CellCODE/	TRUE	matrix factorisation	R, C++, Fortran	4	100	TRUE	
CoD	RNA-seq	supervised	https://doi.org/10.1101/232079	2018	Micro blood disease induction	https://www.pitt.edu/~mhilka/CellCODE/	FALSE	regression	web tool	32	8,00	TRUE	
UNDO	MA	unsupervised	https://doi.org/10.1101/232080	2018	Cancer transcriptome	https://www.pitt.edu/~mhilka/CellCODE/	TRUE	matrix factorisation	R	16	5,50	TRUE	
ESTIMATE	MA + RNA-seq	supervised	https://doi.org/10.1101/232081	2018	Tissue mixtures	https://www.pitt.edu/~mhilka/CellCODE/	TRUE	enrichment	R	268	44,33	TRUE	
DecorINASeq	RNA-seq	supervised	https://doi.org/10.1101/232082	2018	Cancer transcriptome	https://www.pitt.edu/~mhilka/CellCODE/	FALSE	regression	R	52	8,67	TRUE	
DIA	MA	supervised	https://doi.org/10.1101/232083	2018	Cancer transcriptome	https://www.pitt.edu/~mhilka/CellCODE/	TRUE	regression	R	59	8,00	TRUE	
DSQuire	MA	supervised	https://doi.org/10.1101/232084	2018	Cancer transcriptome	https://doi.org/10.1101/232084	TRUE	probabilistic	matlab	44	7,33	TRUE	
DeMix	MA	supervised	https://doi.org/10.1101/232085	2018	Cancer purity	https://cdna.mit.edu/bwang/DeMix.html	TRUE	probabilistic	C, R	38	6,33	TRUE	
NanoVisor	MA	supervised	https://doi.org/10.1101/232086	2018	Chronic myeloid disease (cell lineage)	https://cdna.mit.edu/bwang/DeMix.html	FALSE	regression	web tool	33	5,00	TRUE	
TIMER	MA + RNA-seq	supervised	https://doi.org/10.1101/232087	2018	Cancer transcriptome	https://cdna.mit.edu/bwang/DeMix.html	FALSE	regression	web tool	33	5,00	TRUE	
Self-directed Method for Cell Type Identification	MA	unsupervised	https://doi.org/10.1101/232088	2018	Cancer transcriptome	NA	TRUE	matrix factorisation	matlab	18	3,00	TRUE	
MMAD	MA	BOTH	https://doi.org/10.1101/232089	2018	in vitro tissue mixtures	https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6105634/	TRUE	regression	matlab	11	183	TRUE	
TBM	RNA-seq	undefined	https://doi.org/10.1101/232090	2018	Defined	https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6105634/	TRUE	probabilistic	unknown	2	0,33	FALSE	
RNAmix	MA	semi-supervised	https://doi.org/10.1101/232091	2018	in vitro tissue mixtures	https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6105634/	TRUE	matrix factorisation	pybind	0	0,00	TRUE	
Statistical expression deconvolution	MA	supervised	https://doi.org/10.1101/232092	2018	NA	https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6105634/	TRUE	matrix factorisation	R	61	871	TRUE	
DSection	MA	supervised	https://doi.org/10.1101/232093	2018	Cytogenetics	https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6105634/	TRUE	probabilistic	octave	33	4,71	TRUE	
decont	MA	unsupervised	https://doi.org/10.1101/232094	2018	Tissue mixtures	https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6105634/	TRUE	enrichment	web tool	31	4,43	TRUE	
Adam-deconvolution	MA	supervised	https://doi.org/10.1101/232095	2018	Infected lung tissue	https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6105634/	TRUE	probabilistic	unknown	95	12,00	TRUE	
ISOLATE	MA	supervised	https://doi.org/10.1101/232096	2018	NA	https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6105634/	FALSE	regression	unknown	76	9,50	TRUE	
Electronical subtraction	MA	supervised	https://doi.org/10.1101/232097	2018	Infected macrophages	https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6105634/	TRUE	probabilistic	pybind	39	4,88	TRUE	
Computational expression deconvolution	MA	supervised	https://doi.org/10.1101/232098	2018	NA	https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6105634/	TRUE	matrix factorisation	R	286	3,98	TRUE	
Robust Computational Reconstruction	MA	supervised	https://doi.org/10.1101/232099	2018	NA	https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6105634/	TRUE	probabilistic	unknown	52	5,78	TRUE	
MiHMM	MA	unsupervised	https://doi.org/10.1101/232100	2018	Synthetic (cell type in silico)	https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6105634/	TRUE	matrix factorisation	R	41	4,56	TRUE	
In silico microdissection	MA	unsupervised	https://doi.org/10.1101/232101	2018	NA	https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6105634/	TRUE	probabilistic	pybind	20	20,00	TRUE	
Mixture models	MA	supervised	https://doi.org/10.1101/232102	2018	NA	https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6105634/	TRUE	probabilistic	R	66	4,40	TRUE	
DECONVOLUTIE	MA	supervised	https://doi.org/10.1101/232103	2018	yeast cell cycle	https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6105634/	TRUE	regression	Java 2	183	8,43	TRUE	
Direct method	MA	unsupervised	https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6105634/	2018	cancer and normal tissue	https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6105634/	TRUE	matrix factorisation	unknown	95	5,33	TRUE	

The most popular language of implementation of published methods is R (49.2 %), followed by Matlab (11.11%), only one tool so far was published in Python.

2.3.1.2 Data type

Also, most of the methods were designed to work with microarray data. There is a high chance that some of them are adaptable to RNA-seq. However, a little number of older methods was tested in a different setup. For some method, as CIBERSORT, demonstrated to work with microarray and applied commonly to RNA-seq by other researchers, the validity of results remains unclear as some studies claim that CIBERSORT performs accurately applied to RNA-seq [?] and other opt against it [? ?]. Most of newer methods (i.e. EPIC [?], quanTlseq [?] or Infino [?]) are specifically designed for RNA-seq TPM-normalized data. Some methods, mostly enrichment-based methods, are applicable to both technologies (i.e. xCell [?]).

2.3.1.3 Objectives of the cell-type deconvolution

It is remarkable that the general aim of the cell-type deconvolution changed with time. The earlier methods aimed to improve the power of differential expression analysis through *purification* of the gene expression. For example, to compare differentially expressed genes (DEG) in T-cell from the blood under two conditions. However, the obtained purified profiles from complex mixtures were often uncertain [?]. Recently, the most mentioned goal of deconvolution is a quantification of proportions of different cell types, especially in the context of cancer transcriptomes motivated by redefinition of immunophenotypes discussed in the previous chapter. The most popular tissue of interest for deconvolution algorithms are cancer tissues and blood. Other applications are cell-cycle time-dependent fluctuations of yeast, brain cells, and glands .

2.3.1.4 Differences between approaches

Mathematically speaking, I have divided methods into four categories: probabilistic, regression, matrix factorisation and convex hull depending on the nature of the approach. Most of the methods (48 - 74.6%) are working within a supervised framework, and only 20% (14) are unsupervised. The approaches will be described in detail in the following section.

There are numerous practical differences between the methods. ?] in their review of deconvolution tools grouped the tools depending on their inputs and outputs. Given the type of outputs, deconvolution can be considered as complete (proportions and cell profiles) or partial (one of those). Moreover, the inputs of the algorithms can be important to evaluate how practical the tool is. The most popular tools and the most recent tools ask for minimal input from the user: the bulk gene expression matrix, or even raw sequencing data [?]. Older methods usually request either at least approximative proportions of mixed cells or purified profiles to be provided. The newer methods include the reference profiles in the tool if necessary. Some tools, including most of purity estimation tools, demand an additional data input as normal samples or another data

type such as CNA data (Timer [?], VoCAL [?]) or image data (quanTIseq [?]). An important parameter is also a number of sources (k) to which the algorithm deconvolutes the mixture. In many methods, it should be provided by the user, which can be difficult in a case of complex mixtures of human tissues. Besides, type of method can also limit the number of sources, for example, a probabilistic framework privilege lower number of sources (2-3) due to the theoretical constraints. In regression depending on provided reference the output number of estimated sources is imposed. Because of the problem of collinearity and similarity of immune cell profiles, it is hard to distinguish between cell sub-types, deconvolution into fine-grain cell subtypes is often called often deep deconvolution. Some methods (i.e., CIBERSORT, Infino, xCell) give specific attention to deconvolution of cell-subtypes. An absolute presence of a cell type in the mixture can also be an essential factor. If it is too low it can reach a detection limit, Electronic subtraction [?] discuss specifically the detection of rare cel-types.

2.3.1.5 Computational efficiency

Running time and the necessary infrastructure are another way to characterize the methods. Although it is hard to compare the running time objectively simultaneously of all the tools because of the heterogeneity of methods and different datasets analysed, some tendencies can be observed. If one thinks about applying deconvolution methods to big cohorts, regression and enrichment-based methods should be well suited. As far as matrix factorisation is concerned, it depends on the implementation (i.e. R vs Matlab) and if the number of sources needs to be estimated (multiple runs for different k parameter) or if a stabilisation needs to applied (multiple runs for the same k parameter). Finally, probabilistic tools seem to be challenging to scale, i.e. authors of Infino admit that their pipeline is not yet applicable at high-throughput.

In order to let user better understand the differences between different mathematical approaches, I will introduce shortly the types of approaches used for cell-type deconvolution of transcriptomes as well as their strong and weak points.

2.3.2 Regression-based methods

Regression models are the most popular methods for bulk gene expression deconvolution. They use estimated pure cell profiles as depending variables (or selected signature genes) that should explain the mixed profiles choosing best β parameters (Eq. (2.3)) that can be interpreted as cell proportions.

A standard type of regression is called linear regression. It reflects linear dependence between independent and dependent variables. The linear regression was developed in the *precomputer age of statistics* [?].

In linear regression, we want to predict a real-valued output Y , given a vector $X^T = (X_1, X_2, \dots, X_p)$. The linear regression model has the form:

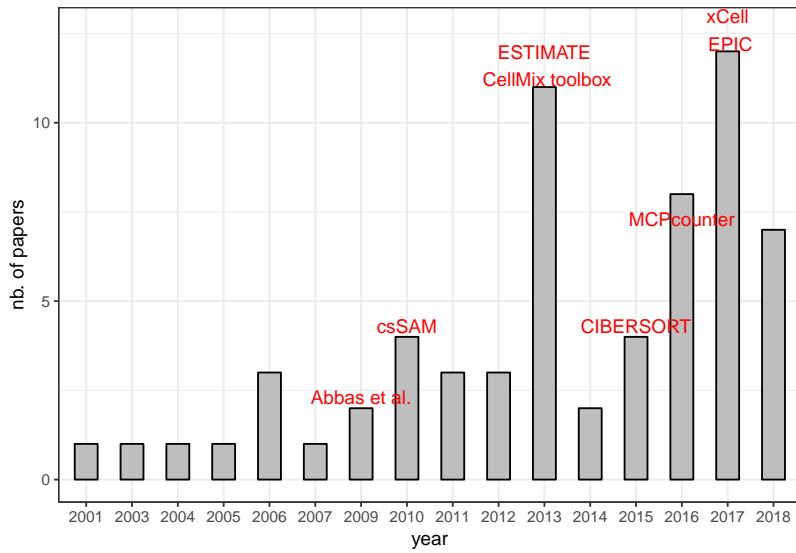


Figure 2.3: Distribution of publications of cell-type deconvolution of bulk transcriptome over the years. In red: hallmark publications. Data gathered based on PubMed and google scholar search in May 2018.

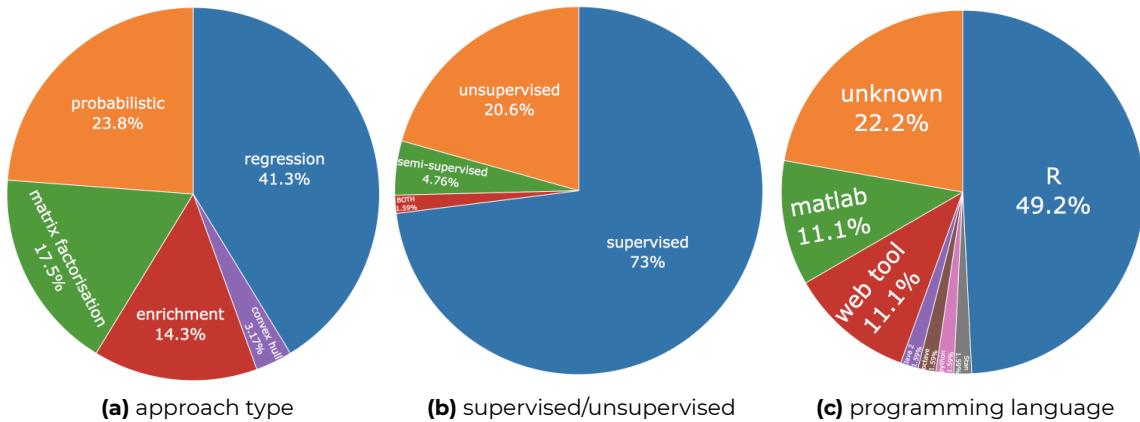


Figure 2.4: Simple statistics illustrating characteristics of published cell-type deconvolution tools: 2.4a - Percentage of used approach type, 2.4b - Percentage of supervised/unsupervised tools, 2.4c - Percentage of the programming languages of implementation. Data gathered based on pubmed and google scholar search in May 2018.

$$f(X) = \beta_0 + \sum_{j=1}^p X_j \beta_j \quad (2.3)$$

Where the β_j s are unknown parameters or coefficients, and X_j s are the explaining variables. Given pairs of $(x_1, y_1), \dots, (x_N, y_N)$, one can estimate coefficients β with an optimization of an objective function (also called cost function).

The most popular estimation method is **least squares**, the coefficients $\beta = (\beta_0, \beta_1, \dots, \beta_n)$ are computed to minimize the residual sum of squares (RSS):

$$RSS(\beta) = \sum_{i=1}^N (y_i - f(x_i))^2 = \sum_{i=1}^N (y_i - \beta_0 - \sum_{j=1}^p x_i \beta_j)^2 \quad (2.4)$$

Ordinary least squares regression is using Eq.(2.4) to compute β .

Ridge regression (Eq.(2.5)) (aka Tikhonov regularization) adds a regularizer (called $L2$ norm) to shrink the coefficients ($\lambda \geq 1$) through imposing a penalty on their size.

$$\hat{\beta}^{ridge} = \underset{\beta}{\operatorname{argmin}} \left\{ \sum_{i=1}^N (y_i - f(x_i))^2 + \lambda \sum_{j=1}^p \beta_j^2 \right\} \quad (2.5)$$

Similarly **Lasso regression** (Equation (2.6)) adds a regularization term to RSS (called $L1$ norm), it may set coefficients to 0 and therefore perform feature selection.

$$\hat{\beta}^{ridge} = \underset{\beta}{\operatorname{argmin}} \left\{ \sum_{i=1}^N (y_i - f(x_i))^2 + \lambda \sum_{j=1}^p |\beta_j| \right\} \quad (2.6)$$

In **Elastic net regression** both penalties are applied.

Support Vector Regression (SVR) is regression using **Supported Vector Machines (SVM)**. In SVR β can be estimated as follows:

$$H(\beta, \beta_0) = \sum_{i=1}^N V(y_i f(x_i)) + \frac{1}{2} \|\beta\|^2 \quad (2.7)$$

where error is measured as follows:

$$V_\epsilon(r) = \begin{cases} 0, & \text{if } |r| < \epsilon, \\ |r| - \epsilon, & \text{otherwise.} \end{cases} \quad (2.8)$$

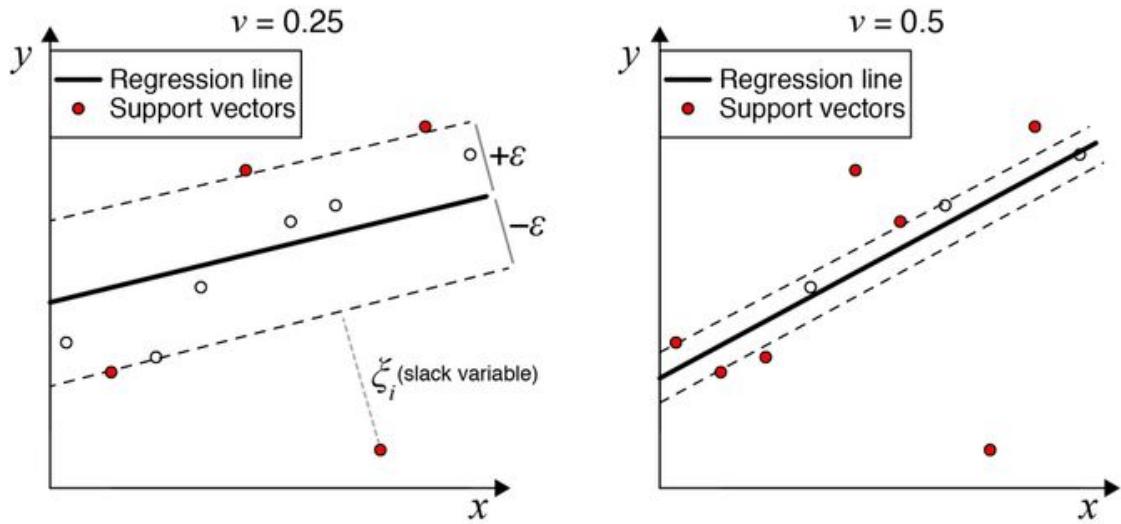


Figure 2.5: Principle of the SVR regression. In SVR regression ϵ represents the limit of error measure, input data points higher than $+\epsilon$ or lower than $-\epsilon$ are called support vectors. The ν parameter in ν -SVR regression controls the distance of training error bonds: left - lower ν value larger bound, right - higher ν margin, smaller bound. Reprinted by permission from Springer Nature [?] © 2018 Macmillan Publishers Limited, part of Springer Nature. All rights reserved.

with ϵ being the limit of error measure, meaning errors of size less than ϵ are ignored.

In the SVM vocabulary, a subset of the input data that determine hyperplane boundaries are called the **support vectors** (Fig.2.5). SVR discovers a hyperplane that fits the maximal possible number of points within a constant distance, ϵ , thus performing a regression.

In brief, in SVR, RSS is replaced by a linear ϵ -insensitive loss function and uses L_2 -norm penalty function. There exist variants of SVR algorithm, i.e. ϵ -SVR [?] and ν -SVR [?]. ϵ -SVR allows to control the error; this favors more complex models. In the ν -SVR the distance of the ϵ margin can be controlled and therefore the number of data points used for regression can be controlled. [?] used an SVM-based method to define cell type-specific genes. A model using ν -SVR with linear kernel was used by [?] in CIBERSORT.

As unconstrained optimization of the objective function can result in negative coefficients, in the context of cell-type deconvolution, authors often aim to avoid as it complicates the interpretation. Therefore, different constraints can be imposed on the β coefficients. The most common conditions are $\beta_0 + \beta_1 + \dots + \beta_n = 1$ and $\forall \beta_i \geq 0$. Solution respecting the non-negativity condition is also called non-negative least squares (NNLS) to contrast with ordinary least squares (OLS). NNLS was adopted by many authors [? ? ? ? ?].

The task can also be solved differently from the computational perspective. [?] and [?] propose to use simulated annealing to minimize the cost function. [?] proposed to solve the task using quadratic programming.

An extensive review on optimization of the objective function for regression methods in cell-type deconvolution was published by [?]. Authors carefully consider different possibilities of parameter choice in the loss and regularization formulations and its performance. They present as well recommendations for construction of basis matrix and data pre- and post-processing. Digital tissue deconvolution (DTD) [?] aims to train the loss function with *in silico* mixtures of single cell profiles resulting in improved performance of rare cell types (present in a small proportion). However, the training is computationally heavy, and the proper training data for bulk transcriptomes are not available.

Since the publication of CIBERSORT [?] some authors [? ?] used the [?] implementation directly with pre/post modifications or with different basis matrix or they re-implemented the SVR regression in their tools [?].

Another recent method EPIC [?] introduced weights related to gene variability. In their constrained regression, they add it explicitly in the cost function modifying RSS (Eq.(2.4)):

$$RSS^{weighted}(\beta) = \sum_{i=1}^N (y_i - \beta_0 - w_i \sum_{j=1}^p x_i \beta_j)^2 \quad (2.9)$$

with the non-negativity and the sum constraints, we discussed above. The w_i weights are corresponding to the variance of the given gene measure in the same cell type. It aims to give less importance to the genes variant between different measurements of the same cell-type. EPIC also allows a cell type that is not a referenced in the signature matrix with an assumption that the non-referenced cell type is equal to 1- a sum of proportions of other cell types (Eq.(2.10)). Authors interpret this non-referenced cell type as the tumor fraction:

$$\beta_m = 1 - \sum_{j=1}^{m-1} \beta_j \quad (2.10)$$

An additional feature of EPIC is advanced data normalization and estimation of mRNA produced by each cell to adjust cell proportions, which was previously proposed by [?] in the context of microarray data:

$$p_j = \alpha \frac{\beta_j}{r_j} \quad (2.11)$$

where p_j are actual cell proportions that are 'normalized' with empirically derived coefficient α and measured r_j is the number of RNA nucleotides in cell type j .

Recently CIBERSORT proposed an *absolute mode* where the proportions are not relative to the leucocyte infiltration but to the sample. It can be obtained with an assumption that the estimation of the proportion of all genes in CIBERSORT matrix is corresponding to sample purity. This functionality was not yet officially published, and it is still in experimental phase [?].

Table 2.2: Contingency table is the count of overlap of genes present in a certain condition (Y) vs. not present (Y-Z) and association to a pathway X (in X or not in X). The contingency table is used in the frequency based test as Fisher exact test.

	Y	Z-Y
in X	a	b
not in X	c	d

Regression methods combined with pre- and post-processing of data can result in estimation of proportions that can be interpreted directly as a percentage of cells in a mixed sample. It is an important feature hard to achieve with other methods. Some methods provide relative proportions of the immune infiltrate [?] and another aim to provide absolute abundance [?]. The absolute proportions are easily comparable between data sets and cell types. Regression-based methods are usually quite fast and can process large transcriptomic cohorts. However, as I will discuss in Validation section, they pose on the hypothesis that the reference profiles available in some context (i.e., blood) are valid in a different one (i.e., tumor) or that profiles extracted from one data type (scRNA-seq) are adapted to deconvolute bulk RNAseq. Most of the recent regression methods focused on estimating proportions and do not estimate context-specific profiles and can process as little as one sample.

2.3.3 Enrichment-based methods

Enrichment-based methods aim to evaluate an amount of activity of a given list of genes within the data. This can be obtained by calculating a score based on gene expression. Traditionally enrichment methods were used to analyze set of DEG. Different statistical approaches were adapted: like Fisher exact test giving a p-value that estimated the chance a given list of genes is over/under present in the input list of DEGs and therefore characterize the condition vs. control expressed genes.

Let's take an example; if one wants to compute enrichment in pathway X of the list of DEG genes Y with the total number of tested genes Z, a contingency table need to be constructed (Tab. 2.2).

In the Fisher exact test formula (Eq. (2.12)) the a, b, c and d are the individual frequencies, i.e. number of genes in of the 2X2 contingency table, and N is the total frequency ($a + b + c + d$).

$$p = \frac{((a+b)!(c+d)!(a+c)!(b+d)!)}{a!b!c!d!N!} \quad (2.12)$$

Another important (>14000 citations) algorithm computing such a score (enrichment score ES) is named gene set enrichment analysis (GSEA) [?] uses sum-statistics. The list of genes user wants to test for enrichment is usually ranked by fold change odd or p-value of DGE analysis.

The high score indicated high activity of genes included in the list. GSEA can also indicate an

anti-activity of correlation. A variant of GSEA, single sample GSEA (ssGSEA) [?] was used by [?], [?] and [?] to compute infiltration scores. In the ssGSEA genes are ranked by their absolute expression. A variance-based variant of GSEA - GSVA [?] was used by [?] for the same purpose. MCPcounter [?] uses an arithmetic mean of gene expression of highly specific signature genes to compute a score.

In this way obtained scores, are not comparable between different cell types and datasets. Therefore some authors propose normalization procedures that make the score more comparable. For instance, xCell uses a platform-specific transformation of enrichment scores. Similarly, Estimate transforms scores for TCGA through an empirically derived formula. MCPcounter authors use z-scoring to minimize platform-related differences. Unfortunately, the normalization is not directly included in the R package

Even though enrichment methods do not try to fit the linear model and derived scores are not mathematically conditioned to represent cell proportions; usually there can be observed a strong linear dependence. An advantage of the enrichment-based methods is the speed and possibility to include distinct signatures that can characterize cell-types and cell-states of different pathways.

2.3.4 Probabilistic methods

The probabilistic methods share a common denominator: they aim to minimise a likelihood function of Bayes' theorem:

$$p(y|\theta) = \frac{p(\theta|y) * p(y)}{p(\theta)} \quad (2.13)$$

In Eq.(2.13) y is our data, θ a parameter, $p(y|\theta)$ posterior, $p(\theta|y)$ likelihood and $p(\theta)$ prior. Prior distribution is what we know about the data before it was generated and combined with a probability distribution of the observed data is called posterior distribution. The likelihood describes how likely it is to observe the data (y) given the parameter θ (probability of y given θ - $p(y|\theta)$). A parameter is characteristic of a chosen model and a hyperparameter is a parameter of prior distribution.

In the literature, there are mainly different types of probabilistic models, one that assumes some type of distribution of mixed sources (i.e., Gaussian or Poisson), others that learn the distribution parameters empirically from a training set, another that try to find the parameters of the distribution given the number of given sources. Then in each case, there are different ways of constructing different priors and posteriors functions. Among used techniques are Markov Chain Monte Carlo or Expectation-Maximisation, which themselves can be implemented in different ways [? ? ? ? , ?].

The probabilistic approaches are the most popular for purity estimation (2 components models), that seems to be possible to extend to 3-components model [?]. As far as cell-type decompo-

sition into a number of cells is concerned, a method published on BioRxiv *Infino* uses Bayesian inference with a generative model, trained on cell type pure profiles. Authors claim their method is notably suited for deep deconvolution that is able to build cell type similarities and estimate the confidence of the estimated proportions which help to interpret the results better.

A probabilistic framework is an attractive approach with solid statistical bases. It can be suited to many specific cases. The pitfalls are (1) the need of prior profiles or correct hypothesis on the distribution parameters (2) reduced performance when applied to high dimensional datasets due to extensive parameters search.

2.3.5 Convex-hull based methods

An emerging family of BSS methods is convex geometry (CG)-based methods. Here, the sources are found by searching the facets of the convex hull spanned by the mapped observations solving a classical convex optimization problem [?]. It can be implemented in many ways [?].

Convex hull can be defined as follows [?]:

*We are given a set P of n points in the plane. We want to compute something called the **convex hull** of P . Intuitively, the convex hull is what you get by driving a nail into the plane at each point and then wrapping a piece of string around the nails. More formally, the convex hull is the smallest convex polygon containing the points:*

- **polygon**: A region of the plane bounded by a cycle of line segments, called **edges**, joined end-to-end in a cycle. Points, where two successive edges meet, are called **vertices**.
- **convex**: For any two points p, q inside the polygon, the line segment pq is completely inside the polygon.
- **smallest**: Any convex proper subset of the convex hull excludes at least one point in P . This implies that every vertex of the convex hull is a point in P .

Convex hull methods have been used in many fields, from economics and engineering, I will discuss it with a focus on biological context to link tightly to cell-type deconvolution.

The central assumptions of Convex hull optimization are that the gene expression of pure cell types is non-negative and that cell type proportions are linearly independent.

The shapes can be fitted to a cloud of points in many ways in order to respond to given optimality criteria. A popular method introduced by [?] and applied to gene expression and morphological phenotypes of biological species employ the **Pareto front** concept which aims to find a set of designs that are the best trade-offs between different requirements.

Visually Pareto front correspond to the edge of the convex hull.

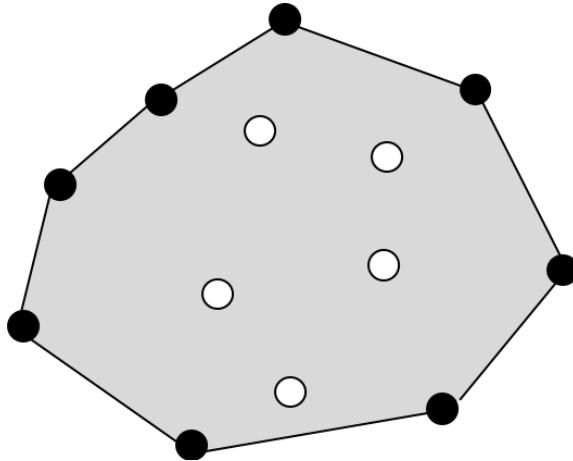


Figure 2.6: Convex hull illustration. A set of points and its convex hull (line). Convex hull vertices are black, and interior points are white. Image reproduced after [?].

[?] proposed Complex Analysis of Mixtures (CAM) method to find the Pareto front (the vertices of X mixed matrix) (a convex set)). In the context of the cell-type deconvolution, it can be said that “the scatter simplex of pure subpopulation expressions is compressed and rotated to form the scatter simplex of mixed expressions whose vertices coincide with cell proportions”[?]. In respect to the assumptions, under a noise-free scenario, novel *marker genes* can be blindly identified by locating the *vertices* of the mixed expression scatter simplex [?]. In the figure (Fig. 2.7), the a_i ’s are cell-type proportions of k cell types, s_i pure cell type expression and x_j mixed expression in sample j . Therefore the vertices correspond to the column vectors of the matrix A (Eq. (2.1)). The genes placed in a distance d from the vertices can be interpreted as marker genes.

In the procedure suggested by [?], before performing CAM, clustering (more precisely affinity propagation clustering (APC)) is applied to the matrix in order to select genes representing clusters, called cluster centers g_m and dimension reduction(PCA) is applied to the sample space. Then in order to fit a convex set, a margin-of-error should be minimized. The Eq. (2.14) explains the computation of the error which computes L_2 norm of the difference between g_m possible vertices and remaining exterior clusters. All possibilities of combinations drew from C_K^M , M number of clusters and K true vertices, are tested.

$$\text{given } \alpha_k \geq 0, \sum_{k=1}^K \alpha_k = 1$$

$$\delta_{m, \{1, \dots, K\} \in C_K^M} = \min_{\alpha_k} \sqrt{g_m - \sum_{k=1}^K \alpha_k g_k} \quad (2.14)$$

Once optimal configuration is found, the proportions are computed using standardised averag-

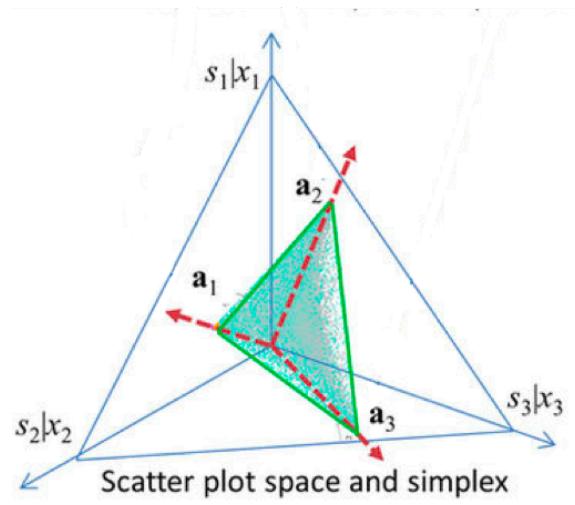


Figure 2.7: Fitting gene expression data of mixed populations to a convex hull shape. The geometry of the mixing operation in scatter space that produces a compressed and rotated scatter simplex whose vertices host subpopulation-specific marker genes and corresponding to mixing proportions.

ing:

$$\hat{\alpha}_k = \frac{1}{n_{\text{markers}}} \sum_{i \in \text{markers}} \frac{x(i)}{\|x(i)\|} \quad (2.15)$$

where $\hat{\alpha}_k$ is proportion of cell type k , n_{markers} number of marker genes (obtained from CAM), and $\|x(i)\|$ is the $L1$ or $L2$ norm of a given marker gene x_i .

Then the cell-type specific profiles are obtained with linear regression. Authors of CAM also propose a minimum description length (MDL) index that determines the number of sources in the mixture. It selects the K minimizing the total description code length.

So far, the published R-Java package CAM does not allow to extract gene specific signatures, and it is not scalable to large cohorts (many samples). In the article, authors apply essential pre-processing steps that are not trivial to reproduce and which are not included in their tool. Authors apply CAM and validate on rather simple mixtures (tissue *in vitro* mixtures and yeast cell cycle).

A slightly different approach was proposed by [?]. It does not require initial dimension reduction steps or clustering before fitting the convex hull, and it is based on a probabilistic framework. The toll *CellDistinguisher* was inspired by topic modeling algorithm [?]. It first computes Q matrix (Eq. (2.16)). Then each row vector of Q is normalized to 1 giving \bar{Q} matrix. Every row of \bar{Q} lies in the convex hull of the rows indexed by the cell-type specific genes. Then L_2 norm of each row is computed. Genes which rows have the highest norm can be used as *distinguishers* or *marker genes*. Then other runs of selections are applied after recentering the matrix to find more markers.

$$Q = XX^T \quad (2.16)$$

Once the set of possible distinguishers is defined, proportions and cell profiles are computed using a Bayesian framework to fit the convex hull. Authors provide a [user-friendly R package *CellDistinguisher*](#). Unfortunately, they do not provide any method for estimation of some sources, which is critical for source separation of complex tissues. Additionally, quantitative weights are provided only for signature genes which number can vary for different sources, and can be as small as one gene. Authors do not apply their algorithm to complex mixtures as tumor transcriptome; they establish a proof of concept with *in vitro* mixtures of tissues.

The convex hull-based method does not require the independence of cell types assumption, nor the non-correlation assumption which can be interesting in the setup of closely related cell types. In theory, they also allow $k > j$ (more sources than samples). So far, the existing tools are not directly applicable to tumor transcriptomes.

2.3.6 Matrix factorization methods

Matrix factorization is a general problem not specific to cell types deconvolution. It has been extensively used for signal processing [?] and extraction of features from images [?]. Matrix factorization can also be called BSS or dimension reduction. Despite quite simple statistical bases they have been proven to be able to solve quite complex problems. Many matrix factorization methods can solve the problem of Eq. (2.1). They can solve it in different ways and concern different hypotheses.

Naturally, matrix factorization methods estimate simultaneously A and S matrices (cell proportions and profiles) given X rectangular matrix (genes \times samples) without any additional input.

2.3.6.1 Principal Components Analysis

One of the most popular methods, **Principal Components Analysis** (PCA) computes projections of the variables onto the space of the eigenvectors of the empirical covariance matrix, the projections are mutually uncorrelated and ordered in variance. The principal components provide a sequence of best linear approximations to that data.

PCA can be computed through eigen decomposition of the covariance matrix. Covariance matrix is computed as follows:

$$\Sigma = \frac{1}{n-1}((X - \bar{x})^T(X - \bar{x})) \quad (2.17)$$

where \bar{x} is mean vector of the feature column in the data X .

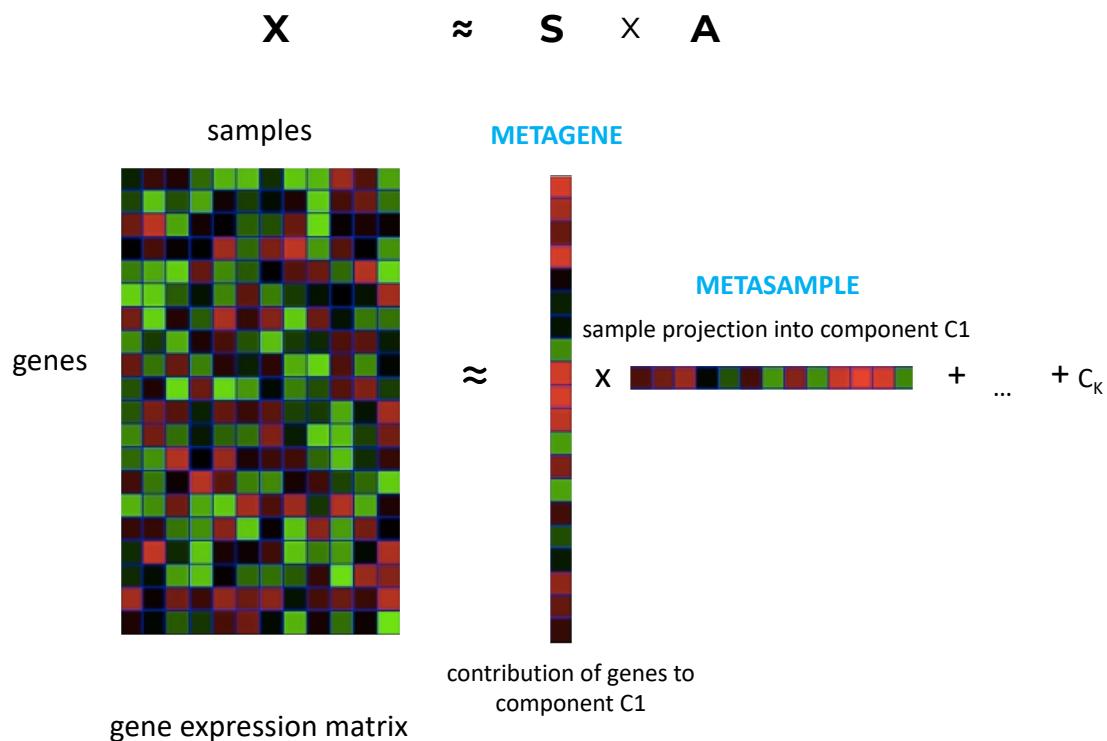


Figure 2.8: Principle of matrix factorisation of gene expression. The gene expression matrix X is decomposed into a set of *metagenes* S matrix and *metasamples* A . Number of components C is defined with parametre k .

Then the matrix is decomposed to eigenvalues:

$$\mathbf{V}^{-1}\boldsymbol{\Sigma}\mathbf{V} = \mathbf{D} \quad (2.18)$$

where \mathbf{V} is the matrix of eigenvectors and the \mathbf{D} diagonal matrix of eigenvalues.

It can be also computed using **singular value decomposition** (SVD) (computationally more efficient way):

$$X = UDV^T \quad (2.19)$$

Here U is an $N \times p$ orthogonal matrix ($U^T U = I_p$) whose columns u_j are called the left singular vectors; V is a $p \times p$ orthogonal matrix ($V^T V = I_p$) with columns v_j called the right singular vectors, and D is a $p \times p$ diagonal matrix, with diagonal elements $d_1 \geq d_2 \geq \dots \geq d_p \geq 0$ known as the singular values. The columns of UD are called the projections of principal components of X on axes.

PCA finds directions in which the samples are dispersed to define Principal Components. This dispersion is measured with variance, and resulting PCA components are variance-ordered.

As nicely described in [?]

The first PC is the vector describing the direction of maximum sample dispersion. Each following PC describes the maximal remaining variability, with the additional constraint that it must be orthogonal to all the earlier PCs to avoid it contains any of the information already extracted from the data matrix. In other words, each PC extracts as much remaining variance from the data as possible. The calculated PCs are weighted sums of the original variables, the weights being elements of a so-called loadings vector. Inspection of these loadings vectors may help determine which original variables contribute most to this PC direction. However, PCs being mathematical constructs describing the directions of greatest dispersion of the samples, there is no reason for the loadings vectors to correspond to underlying signals in the dataset. Most of the time, PCs are combinations of pure source signals and do not describe physical reality. For this reason, their interpretation can be fraught with danger.

Especially in the context of the cell-type deconvolution, it can imagine that different cell-types contribute to the variance, but one PC could explain the joint variance of many cell types.

[?] used SVD to compute matrix inversion in order to separate tumor from the stroma. The method was applied to tumor transcriptomes and gives purity estimation quite different from other popular enrichment-based method ESTIMATE [?].

[?] in CellMapper uses a semi-supervised approach based on SVD decomposition to dissect human brain bulk transcriptome. Authors define a query gene (a specific known gene), and then

they decompose transcriptome into components (eigenvectors) and multiply by weights that are higher for the components correlated with the query gene. Then the matrix is transformed back to gene \times samples matrix, but query signal is amplified. The point is to find marker genes that characterize the same cell-type as the query gene. Authors did not aim at the identification of cell-type proportions or cell types profiles but identification of cell-type specific markers. They underline applicability of the method to rare cell types where many markers are not available. This approach was proposed by authors to be used to prioritize candidate genes in disease susceptibility loci identified by GWAS.

2.3.6.2 Non-negative matrix factorisation

Non-negative matrix factorization [?] is an alternative approach to principal components analysis. It requires data are non-negative and it estimates components that are non-negative as well. It finds its application in image analysis and gene expression analysis where analyzed data are non-negative. The $N \times p$ data matrix X is approximated by

$$X \approx WH$$

where W is $N \times r$ and H is $r \times p$, $r \leq \min(N, p)$. We assume that $x_{ij}, w_{ik}, h_{kj} \geq 0$.

Which is a special case of Eq. (2.1).

The matrices W and H are found by maximizing

$$\mathcal{L}(W, H) = \sum_{i=1}^N \sum_{j=1}^p [x_{ij} \log(WH)_{ij} - (WH)_{ij}] \quad (2.20)$$

The log-likelihood from a model in which x_{ij} is drawn from a pre-defined distribution (i.e., Poisson) with a mean $(WH)_{ij}$.

This formula can be maximized through minimization of divergence:

$$\min_{W,H} f(W, H) = \frac{1}{2} \|X - WH\|_F^2 \quad (2.21)$$

Where $\|\cdot\|_F$ is Frobenius norm, which can be replaced by Kullback-Leibler divergence.

The optimization can be done employing different methods:

- **euclidean** update with multiplicative update rules, it is the classic NMF [?]

$$\begin{aligned} W &\leftarrow W \frac{XH^T}{WHH^T} \\ H &\leftarrow H \frac{W^TX}{W^TWH} \end{aligned} \quad (2.22)$$

- **alternating least squares** [?] where the matrices W and H are fixed alternatively
- **alternating non-negative least squares** using projected gradients (??)
- **convex-NMF** [?] imposes a constraint that the columns of W must lie within the column space of X , i.e. $W = XA$ (where A is an auxiliary adaptative weight matrix that fully determines W), so that $X = XAH$. In this method only H must be non-negative.

The NMF algorithms can differ in initialization method as well and even in situations where $X = WH$ holds exactly, and the decomposition may not be unique. This implies that the solution found by NMF depends on the starting values. The performance of different combinations applied to MRS data from human brain tumors can be found in [?].

[?] created an NMF Matlab toolbox and demonstrated applicability of NMF (using Kullback-Leibler divergence and euclidean multiplicative update [?] to cancer transcriptomes with focus on cancer subtyping (focusing on the H matrix). [?] also proposed a way to evaluate the optimal number of factors (sources) to which matrix should be decomposed.

NMF as imposing non-negativity in the context of decomposition of transcriptomes seems as an attractive concept as both cell profiles and cell proportions should be non-negative. It is not surprising then that some authors used NMF to perform cell-type deconvolution.

To my knowledge, *deconf* [?] was the first tool proposing NMF cell-type deconvolution of PBMC transcriptome, of considerable dimensions, 80 samples (40 control and 40 cases) of Tuberculosis. [?] employed random initialization and alternating non-negative least squares to minimize the model divergence. The complete deconvolution of the transcriptome was used to perform DEG analysis on the deconvoluted profiles.

[?], not only presented exhaustive literature review through implementing cell-type deconvolution methods in an R package *CellMix* [?] but also proposed a semi-supervised NMF for cell-type deconvolution and published an R package implementing different NMF methods [?]. The semi-supervised version of NMF proposed by [?], need a set of specific marker genes for each desired cell type. Then at initialization and after each iteration of the chosen NMF algorithm (applies to some versions of NMF [? ? ?]), “each cell type signature has the values corresponding to markers of other cell types set to zero. The values for its own markers are left free to be updated by the algorithm’s own iterative schema”. Applying their algorithm to *in vitro* controlled dataset [GSE11058 [?]], testing selected NMF implementations and a varying number of markers per cell, authors observed the best performance with guided version of *brunet* [?] implementation.

[?] applied NMF to separate tumor from stroma in pancreatic ductal carcinoma (PDAC) using multiplicative update NMF. They scaled H matrix rows to 1 so that the values correspond to the proportions. Authors tested different possibilities of the number of sources (k), the final number

of factors was defined through hierarchical clustering on gene-by-gene consensus matrix of top 50 genes of each component.

Finally [?] proposed post-modified NMF in order to separate *in vitro* mixtures of different tissues.

In brief, NMF is a popular, in biology, algorithm performing source separation with non-negativity constraint. It was applied to *in vitro* cell-mixtures and blood transcriptomes, showing a satisfying accuracy of cell-type *in silico* dissection and evaluating proportions. It was also applied in cancer context. However, it did not recover cell-type specific signals but rather groups of signals that could be associated with cancer or stroma.

2.3.6.3 Independent Components Analysis

Independent Components Analysis is written as in Eq. (2.1) maximizing *independence* and *non-Gaussianity* of columns of S : S_i . It was first formulated by [?]

The independence can be measured with entropy, kurtosis, mutual information or negentropy measure $J(Y_j)$ [?] defined by

$$J(Y_j) = H(Z_j) - H(Y_j) \quad (2.23)$$

where $H(Y_j)$ is entropy, Z_j is a Gaussian random variable with the same variance as Y_j . Negentropy is non-negative and measures the deviation of Y_j from Gaussianity. An approximation of negentropy is used in popular implementation of **FastICA** [?]. Other existing implementations of ICA are

- Infomax [?] using Information-Maximization that maximizes the joint entropy
- JADE [?] on the construction of a fourth-order cumulants array from the data

However, they are usually a lot slower which limits their application to a large corpus of data and [?] demonstrated that FastICA gives the most interpretable results.

Therefore, I will focus on FastICA implementation as it will be extensively used in the Results part.

FastICA requires *prewhitening* of the data (centering and whitening). Centering is removing mean from each row of X (input data). Whitening is a linear transformation that columns are not correlated and have variance equal to 1.

Prewhtenning

1. Data centering

$$x_{ij} \leftarrow x_{ij} - \frac{1}{M} \sum_{j'} x_{ij'} \quad (2.24)$$

x_{ij} : data point

2. Whitenning

$$\mathbf{X} \leftarrow \mathbf{ED}^{-\frac{1}{2}} \mathbf{E}^T \mathbf{X} \quad (2.25)$$

Where \mathbf{X} - centered data, \mathbf{E} is the matrix of eigenvectors, \mathbf{D} is the diagonal matrix of eigenvalues

Algorithm 1 FastICA multiple component extraction

Input: K Number of desired components

Input: $X \in \mathbb{R}^{N \times M}$ Prewhitened matrix, where each column represents an N -dimensional sample, where $K \leq N$

Output: $A \in \mathbb{R}^{N \times K}$ Un-mixing matrix where each column projects \mathbf{X} onto independent component.

Output: $S \in \mathbb{R}^{K \times M}$ Independent components matrix, with M columns representing a sample with K dimensions.

(2.26)

```

1: for  $p \leftarrow 1, K$  do
2:    $\mathbf{w}_p \leftarrow$  Random vector of length  $N$ 
3:   while  $\mathbf{w}_p$  changes do
4:      $\mathbf{w}_p \leftarrow \frac{1}{M} X g(\mathbf{w}_p^T X)^T - \frac{1}{M} g'(\mathbf{w}_p^T X) \mathbf{1} w_p$ 
5:      $\mathbf{w}_p \leftarrow \mathbf{w}_p - (\sum_{j=1}^{p-1} \mathbf{w}_p^T \mathbf{w}_j \mathbf{w}_j^T)^T$ 
6:      $\mathbf{w}_p \leftarrow \frac{\mathbf{w}_p}{\|\mathbf{w}_p\|}$ 
7:   end while
8: end for
9: where  $\mathbf{1}$  is a column vector of 1's of dimension  $M$ 
Output:  $A = [\mathbf{w}_1, \dots, \mathbf{w}_K]$ 
Output:  $S = \mathbf{A}^T \mathbf{X}$ 

```

However, the results of this algorithm (Alg. 1. (2.26)) are not deterministic, as the \mathbf{w}_p initial vector of weights is generated at random in the iterations of fastICA. If ICA is run multiple times, one can measure **stability** of a component. Stability of an independent component, regarding varying the initial starts of the ICA algorithm, is a measure of internal compactness of a cluster of matched independent components produced in multiple ICA runs for the same dataset and with the same parameter set but with random initialization [?].

The Icasso procedure can be summarized in a few steps :

1. applying multiple runs of ICA with different initializations
2. clustering the resulting components
3. defining the final result as cluster centroids
4. estimating the compactness of the clusters

In brief, ICA looks for a sequence of orthogonal projections such that the projected data look as

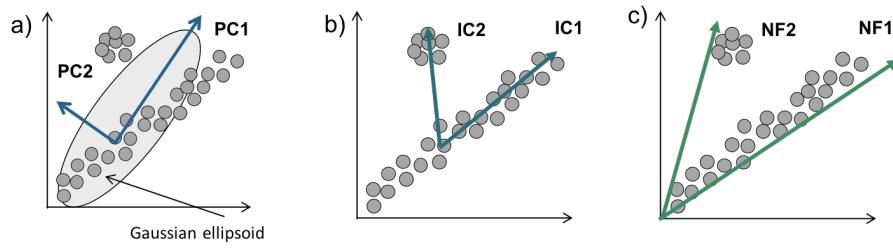


Figure 2.9: Simple illustration of matrix factorisation methods. Adapted with permission from [?]

far from Gaussian as possible. ICA starts from a factor analysis solution and looks for rotations that lead to independent components.

So far, ICA was used to deconvolute transcriptomes into biological functions [? ? ? ? ?]. However, it has never been used for cell-type deconvolution.

In theory, ICA outputs: S could be interpreted as sources and A as proportions in the cell-type deconvolution context. In practice, the fact that the ICA allows negative weights of projections, it makes the interpretation less trivial.

To my knowledge, my DeconICA R-package (that will be described in the results part) is the first method allowing interpretation of ICA-based signals as cell-type context-specific signatures and quantify their abundance in the transcriptome.

All in all, matrix factorization methods are able to decompose a gene expression matrix into a weighted set of genes (metagene)(S) and weighted set of samples (metasample A). Discussed here PCA, NMF and ICA differ in constraints and starting hypotheses. PCA components are ordered by variance and are orthogonal in the initial space of data (Fig. 2.9). NMF impose non-negativity constraint and ICA independence of sources hypothesis. Components of NMF and ICA are not ordered. For all the matrix factorization methods number of components (or factors) (k) needs to be given to the algorithm. Some authors propose a way to estimate the optimal number of components usually justified in a specific context. NMF and SVD were applied in the context of cell-type deconvolution while ICA, so far, was used to dissect transcriptome into factors related to signaling pathways, technical biases or clinical features. Also, ICA was proven to find reproducible signals between different datasets [? ?]. I am going to discuss this aspect of the Results section.

2.3.7 Attractor metagenes

A method proposed by [?] that can be run in semi-supervised or unsupervised mode is called attractor metagenes. Authors describe their rationale as follows:

We can first define a consensus metagene from the average expression levels of all genes in the cluster, and rank all the individual genes in terms of their associ-

ation (defined numerically by some form of correlation) with that metagene. We can then replace the member genes of the cluster with an equal number of the top-ranked genes. Some of the original genes may naturally remain as members of the cluster, but some may be replaced, as this process will “attract” some other genes that are more strongly correlated with the cluster. We can now define a new metagene defined by the average expression levels of the genes in the newly defined cluster, and re-rank all the individual genes concerning their association with that new metagene; and so on. It is intuitively reasonable to expect that this iterative process will eventually converge to a cluster that contains precisely the genes that are most associated with the metagene of the same cluster so that any other individual genes will be less strongly associated with the metagene. We can think of this particular cluster defined by the convergence of this iterative process as an “attractor,” i.e., a module of co-expressed genes to which many other gene sets with close but not identical membership will converge using the same computational methodology.

Which in pseudocode works as described in Algorithm 2 (2.27) and it is implemented in R code is available online in [Synapse portal](#).

Algorithm 2 Attractor metagenes algorithm

Input: α shrinkage parameter

Input: $X \in \mathbb{R}^{N \times M}$ gene expression matrix

Output: m_j metagene of g_{seed}

(2.27)

```

 $g_{seed} \leftarrow a \text{ gene from } 1 : N$ 
2:  $I^\alpha(g_{seed}; g_i)$                                  $\triangleright \text{compute association between } g_{seed} \text{ and } g_i$ 
    $w_i = f(I^\alpha(g_{seed}; g_i))$                    $\triangleright \text{compute weights for each gene}$ 
4:  $m_0 = \frac{\sum_{i=1}^N g_i w_i}{\sum_{i=1}^N w_i}$        $\triangleright \text{compute metagene as weighted average of all genes}$ 
    $I^\alpha(m_0; g_i)$                              $\triangleright \text{compute association between metagene } m_0 \text{ and each gene } g_i$ 
6: repeat
    $w_i = f(I^\alpha(m_0; g_i))$ 
8:    $m_j = \frac{\sum_{i=1}^N -1_{i=1}(m_0 w_i)}{\sum_{i=1}^N -1_{i=1} w_i}$ 
until  $m_{j+1} = m_j$ 
  
```

The produced signatures' weights are non-negative. In the original paper, the generation of tumor signatures leads to three reproducible signatures among different tumor types, including leucocyte metagene. Typically with the essential parameter $\alpha = 5$, they discovered typically approximately 50 to 150 resulting attractors.

This method was further to study breast cancer [?] and to SNP data [?].

There is a possibility to tune the α parameter in order to obtain more or less metagenes that would be possibly interpretable as cell-type signatures.

2.3.8 Others aspects

Here I will discuss transversal aspects common to most deconvolution methods. They play the critical role in the final results and are often omitted while algorithms are published which impacts the reproducibility significantly.

2.3.8.1 Types of biological reference

Let us consider the case of the deconvolution where neither A or S are not known (Eq. (2.1)) (as in the case of cancer transcriptomes), and we would like to estimate cell proportions or both cell proportions and cell profiles. No matter if the method is supervised or unsupervised at some point of the protocol the biological knowledge about cell types is necessary in order to either derive the model or interpret the data. I discussed signatures from the biological perspective in Section 1.3.3. Here, I would like to stress the importance of the design of gene signatures which aim is to facilitate cell-type deconvolution.

Depending on chosen solution different type of reference can be used. In regression algorithms, a way to approximate the individual cell-type profiles is necessary to estimate proportions. However, the genes that are the most variant between cell types are enough for regression, and not all profiles are necessary. A matrix containing gene expression characteristic for a set of cell-types, often for selected, most discriminative, genes is called **basis matrix**. The choice of the genes and the number of the genes impact the outcome [?] significantly. Therefore, most of the regression methods come together with a new basis matrix, ranging from hundreds to tens of genes. Typically, genes selected for basis matrix should be cell-type specific in respect to other estimated cell types, validated across many biological conditions [?]. [?] adds a weight directly in the regression formula (see Eq. (2.9)) that corresponds to the variability of a signature gene between independent measurements of the same cell type so that the least inter-cell type variable genes have more weight in the model. *CellMix* [?] regroups different indexes to select the most specific genes based on signal-to-noise ratio. However, the most popular method is the selection of differentially expressed genes between pure populations. Often criteria for the optimal number of genes in the basis matrix are not knowledge-based but data-driven. [?] uses matrix mathematical property - condition number of basis matrix (*kappa*) in order to select the number of genes. CIBERSORT and many other regression methods follow the same approach. [?] also added another step while constructing the basis matrix, and it preselects reference profiles having maximal discriminatory power. Some authors [? ?] propose to find marker genes through correlation with a provided marker gene (a single one or a group of genes).

In enrichment methods, **gene list** can be enough to estimate cell abundance, sometimes (i.e., GSEA) ranked gene list is necessary. The choice of extremely specific markers is crucial for accurate cell-type abundance estimation. The choice of markers can also be platform-dependent,

this point is strongly underlined in [?]. An interesting possibility is the use of gene list of different *cell states* in order obtain coarse-grain resolution.

The impact of missing gene from a signature in the bulk dataset remains an unanswered question. It would be logical that shorter the gene list for a specific cell, a lack of a gene can have more impact on the result. There is a need for an accurate threshold between robustness and accuracy of the method.

In unsupervised methods, purified cell-profiles, signatures or list of genes can be used ***a posteriori*** to interpret the obtained sources. Even though the choice of reference does not affect the original deconvolution, it affects the interpretation. The advantage of *a posteriori* interpretation is a possibility to use different sources and types of signatures in order to provide the most plausible interpretation. It is common that the way of interpretation of components is not included in the deconvolution tool [? , ?], even though it is a crucial part of the analysis.

For the deconvolution of tumoral biopsies, most of the reference profiles, up to now, are coming from the blood, which is the most available resource. Therefore most of the methods make a hypothesis that blood cell-type profiles/signatures are a correct approximation of cancer infiltrating immune cells. Rare models like PERT [?] or ImmuneStates [?] discuss the perturbation of the blood-derived profiles in diseases.

With the availability of single-cell RNA-seq of human cancers [? ? ? ? ? ? ?], we gain more knowledge on immune cells in TME, and there is growing evidence that they differ importantly from blood immune cells. [?] show that lymph node-resident immune cells have expression profile closer to blood immune cells than cancer immune cells. [?] shows, using a synthetic bulk dataset that using single cell profiles with existing regression methods (CIBERSORT) can improve their performance in the cancer context. However, availability of scRNA-seq remains succinct and probably do not embrace the patient heterogeneity that can be found in large bulk transcriptome cohorts.

2.3.8.2 Data processing

Data pre- and post-processing can have a substantial impact on the deconvolution. Many authors apply strong filtering of genes [?], removing probes with the lowest and the highest expression (potential outliers). In many cases, data preprocessing is not detailed and therefore impossible to reproduce.

There is also a debate on the data normalization.

In microarray analysis pipeline, data are transformed into log-space, usually $\log_2(x + 1)$ in order to make the distribution closer to Gaussian and therefore facilitate statistical hypothesis testing [?]. It also stabilizes the variance [?].

Most of the authors suggest to use counts (not transformed into log-space) for estimating cell abundance as log-transformed data violate the linearity assumption [?], some opt against it [? ?], and some envisage both possibilities [? ?].

For the RNA-seq data TPM (transcripts per million) normalization is preferred or even required by most methods [? ? ?]. [?] performed an extensive analysis of RNA-seq data processing that preserves best the linearity for deconvolution suggesting that TPM normalization for deconvolution studies.

For matrix factorization: PCA, ICA the data are usually log-transformed and centred to reduce the influence of extreme values or outliers. NMF for cell-type deconvolution is usually applied in non-log space.

2.3.8.3 Validation

Most of algorithm validation starts with *in silico* mixtures (Fig. 2.10). In published articles, the bulk transcriptome is simulated in two ways (1) mixing numerically simulated sources at defined proportions of given distribution (i.e. uniform) using linear model (for instance NMF) (2) using sampling (for instance Monte Carlo) to randomly select existing pure profiles and mixing them (additive model) at random proportions. To the obtained bulk samples, noise can be added at different steps of the simulation. Additional parameters can be defined in *in silico* mixtures, for instance, CellMix allows defining the number of marker genes (specific to only one source) for each cell type. The simulated benchmark based on single cell data was used in [?] and [?]. In this framework, simulated data was obtained through summing single cell profiles at known proportions. The main pitfall of those methods is that in the proposed simulations the gene covariance structure is not preserved. In reality, the proportions of cell types are usually not random, and some immune cell types can be correlated or anti-correlated. In addition, these simulations create a simple additive model which perfectly agrees with the linear deconvolution model. This is probably not the case of the real bulk data affected by different factors as cell cycle, technical biases, patients heterogeneity and especially cell-cell interactions.

Naturally, algorithms validated with simulated mixtures are then validated with controlled *in vitro* mixtures of cell types or tissues mixed in known proportions. The most popular benchmark datasets are:

- mix of human cell lines Jurkat, THP-1, IM-9 and Raji in four different concentration in triplicates and the pure cell-line profiles ([GSE11058](#)) [?];
- mix of rat tissues: liver, brain, lung mixed in 11 different concentrations in triplicates and the pure tissues expression([GSE19830](#)) [?]

Similar simple mixtures are also proposed by other authors [? , ?]. This type of benchmark adds the complexity of possible data processing and experimental noise. However, it still follows an almost perfect additive model as the cell/tissues do not interact and they are only constituents of the mixture.

Several tools performed systematic benchmark using PBMC or whole blood datasets, where for a number of patients (that can be over one hundred) FACS measured proportions of selected cell types and bulk transcriptomes are available. Many such datasets can be found at [IMMPORT](#)

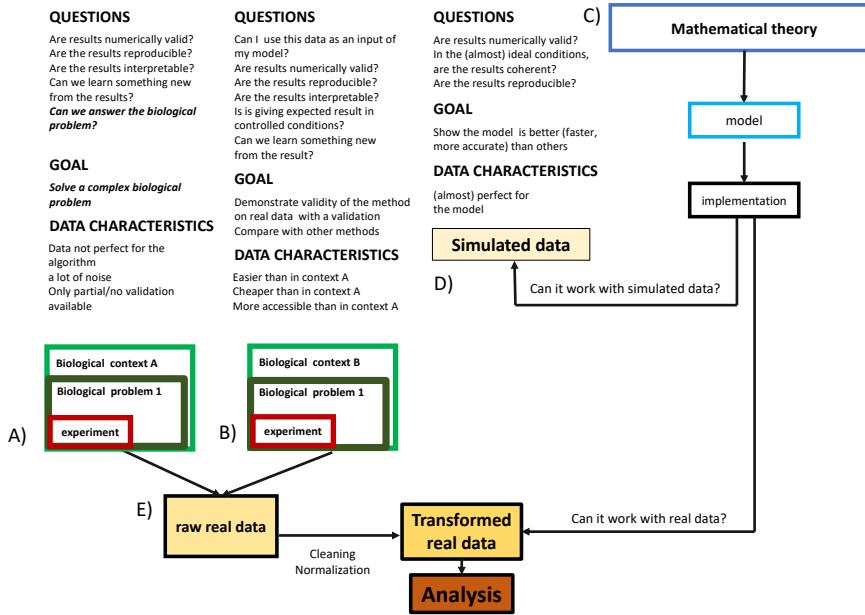


Figure 2.10: From theory to practice: a simplified pipeline of model validation. The scheme reflects pipeline of data validation commonly used in transcriptome deconvolution methods validation. The project can be started from a biological problem (A) and then a way to solve the problem in mathematical model (C) is tested. Most commonly, the project starts with the model (C) then it is tested on simulated data (D). Next level of difficulty is testing the model with real data, so-called, benchmark data (B) that were generated in some biological context different from initial problem. They need to be usually normalized (E) before the model is challenged. B data are widely used as they are easily available and there is some validation available facilitating comparison. Lastly, it is assumed that if the method works fine in the context B, it will work as well in the context A, preferably accompanied by some partial validation. One can replace A by cancer transcriptomics, B by blood data or *in vitro* mixtures if the focus is TME bulk transcriptomics deconvolution.

database. [?] kindly shared with scientific community two datasets with a considerable number of patients (~ 80 and ~ 110) and processed FACS data (actual proportions) on their [github repository](#). It is still important to remember that liquid tissues are easier to deconvolute and for the tools using *a priori* reference, the reference profiles are obtained from the blood. Therefore the context remains consistent.

For the solid cancer tissues deconvolution, some of the tools were validated with the stained histopathology cuts using *in situ*-hybridisation (ISH) [?, ?] or immunohistochemistry (IHC) [?, ?]. Often this method estimates a limited number of cell types and the measured abundance of pictures can also be biased by the technical issues (image/ staining quality).

Authors of EPIC validated their tool with paired RNAseq and Lymph node FACS-derived proportions in 4 patients ([GSE93722](#)). They also noticed that it is more straightforward to correctly evaluate lymph node immune cell types than cancer infiltrating cell types as lymph node-resident cells are more similar to the blood immune cells.

FACS and gene expression of blood cancer (Follicular lymphoma) were also used by [?] for 14 patients (unpublished data). For solid tissues, [?] used paired FACS and expression datasets of normal lung tissues for B-cell and CD8 and CD4 T cells of 11 patients (unpublished data).

Some authors proposed to cross-validate estimated proportions with estimated proportions based on a different data input (i.e., methylome) [?, ?, ?] or CNA [?]. This type of validation is interesting, even though in many projects only one type of data types are available for the same samples. TCGA data is one of few exceptions.

Finally, a validation of deconvolution of solid cancer tissues remains incomplete as no paired expression and FACS data is available up to date.

2.3.8.4 Statistical significance

A little number of tools propose a statistical significance assessment. CIBERSORT computes empirical p-values using Monte Carlo sampling. Infino authors [?] provide a confidence interval for the proportion estimations. This allows knowing which proportion estimation are more trustful than other.

Most tools compare themselves to others measuring the accuracy of the prediction, or Pearson correlation, on the benchmark datasets (described above). Often, in the idealized mixtures, methods perform well. Evaluation of their performance in cancer tissues remains unanswered without proper statistical evaluation.

2.3.9 Summary

The field of computational transcriptome deconvolution is continuously growing. Initially used to solve simple *in vitro* or simulated mixtures of quite distinct ingredients, then to deconvolute

blood expression data, finally applied to solid cancer tissues. In cancer research, digital quantification of cancer purity becomes a routine part of big cancer research projects [?]. Cell-type quantification, even though the validation framework and statistical significance of deconvolution tools can still be improved, seems to be considered as a popular part of an analytical pipeline of bulk tumor transcriptomes [?]. Different types of approaches try to solve the deconvolution problem, focusing on different aspects of the quantification, or proposing methodologically different approaches. Methods proposing an unsupervised solution to the deconvolution problem of transcriptomes are still underrepresented. All the tools assume a linear additive model without explicitly including the impact of possible interactions on the cell-type quantification. The tools that met the most prominent success were proven by the authors to be readily applicable to a variety of cancer datasets and reusable without an extra effort (through a programming library or web interface). The field is still waiting for a gold standard validation benchmark that would allow a fair comparison of all the tools in solid cancer tissues. It is also remarkable that the recent methods focus on quantification of the abundance of an average representation of cell-types without aspiring to deconvolute the cell-type context-specific profiles. Thanks to various cancer single-cell data and big-scale projects [?], we will be able to improve the existing deconvolution approaches and finally replace the collection of bulk transcriptomes by a collection of scRNA-seq ones.

2.4 Deconvolution of other data types

The transcriptome data is not the unique omic data type that can be used to infer cell type proportions. Genomic and epigenomic data was used in numerous publications to perform cell-type deconvolution or estimate sample purity. I will present a general landscape of the tools and methods used for this purpose.

2.4.1 DNA methylation data

Cell-type composition can be computed from DNA methylation data (described in Section X). In EWAS (Epigenome Wide Association Studies) variation origination from cell types is considered as an important confounding factor that should be removed before comparing cases and controls and defining Differentially Methylated Positions (DMPs). [?] reviewed ten tools for epigenome deconvolution. Authors identify six of the described methods as reference-free (which I called in this Chapter *unsupervised*), three are regression-based, and one is semi-supervised. Another review on this topic was authored by [?].

Unsupervised methods employed in methylome cell-type deconvolution are RefFreeEWAS [?], SVA [?] are based on SVD, ISVA based on ICA [?]) are more general methods that aim to detect and remove confounders from the data (that do not need to be necessary the cell types). RUVm [?] is a semi-supervised method using generalized least squares (GLS) regression with negative reference also used to remove *unwanted variation* from the data and could be potentially

adapted to cell-type deconvolution. EWASher [?] is linear mixed model and PCA based method that corrects for cell-type composition. Similarly, ReFACTOr [?] use sparse PCA to remove the variation due to cell-type proportions. [?] proposed RefFreeCellMix: an NMF model with convex constraints to estimate factors representing cell types and cell-type proportions and a likelihood-based method of estimating the underlying dimensionality (k number of factors). A different tool MeDeCom [?] uses alternating non-negative least squares to fit a convex hull.

As far as supervised methods are concerned, EPiDISH (Epigenetic Dissection of Intra-Sample-Heterogeneity) R-package [?] includes previously published tools: Quadratic programming method using reference specific DNase hypersensitive sites [Constrained Projection (CP) [?]], adapted to methylome deconvolution CIBERSORT algorithm (ν -SVR) and robust partial correlations (RPC) method (a form of linear regression). Reference cell-type specific profiles were obtained from the blood.

eFORGE [?] can detect in a list of DMPs if there is a significant cell-type effect.

EDec [?] uses DNAm to infer factors proportions using NMF and then derives factors profiles though linear regression of transcriptome data of cancer datasets. Authors identify the tumor and stroma compartments and profiles. However, they admit the error rate for profiles is quite high for most genes.

[?] focused on intra-tumor heterogeneity (clonal evolution) based on DNAm data. Profiles obtained from cell lines were used in a regression model to identify the proportions of sub-clones in breast cancer data. InfiniumPurify [?] and LUMP [?] uses DNAm to estimate sample purity.

The validation framework for methylation deconvolution is very similar to transcriptome ones: in silico mixtures and FACS-measured proportions of the blood. Most of the tools assume the cell composition is a factor the most contributing to the variability and therefore SVD/PCA based approaches are sufficient to correct for the variability. According to [?], this assumption was not proven to hold true in solid tissues like cancer. Supervised methods have the same drawback as in the case of the transcriptome, and they use purified profiles from one context to derive cell proportions in a different context. In overall, it seems that no study proposed cell-type quantification based on methylome profiles in a pan-cancer manner.

2.4.2 Copy number aberrations (CNA)

To my knowledge there is no method using CNA data in order to estimate cell-type composition, as CNA occur in tumor tissue and natural distinction can be made between tumor and normal cells and within tumor cells (intra-tumor).

Therefore, copy number aberrations can be used to estimate tumor purity and clonality. BACOM 2.0 [?], ABSOLUTE [?], CloneCNA [?], PureBayes [?], CHAT [?], ThetA [?], SciClone [?], Canopy [?], PyClone [?], EXPANDS [?] estimate tumor purity and quantify true copy numbers. [OmicTools website](#) reports 70 tools serving this purpose and their review goes beyond the scope of my work.

Most tools use tumor and normal samples, paired if possible. [QuantumClone](#) seem to be the only tool that requires a few samples from the same tumor (in time or space dimension).

[?] published Consensus measurement of purity estimation that combines purity estimations based on different data types (available in [cBioportal](#)) using: ESTIMATE [?] (gene expression data), ABSOLUTE [?] (CNA), LUMP [?] (DNAm and IHC of stained slides). Authors concluded that the estimation based on different data types highly correlate with each other, besides the IHC estimates, which suggests that IHC provides a qualitative estimation of purity.

2.5 Summary of the chapter

A plethora of machine learning solutions has been developed to solve problems of different nature. Supervised and Unsupervised approaches can be distinguished depending if a model is provided a set of training data with known response or the algorithm works blindly trying to find patterns in the data. Some of the algorithms found an essential application in healthcare and are included in the clinical routine.

One of the critical problems that can, in theory, be solved with ML, is bulk omic data deconvolution. Different types of deconvolution of cancer samples can be distinguished: clonal, purity and cell-type deconvolution. Here I focused on cell-type deconvolution of transcriptome data. Through an extensive review, I presented 64 tools and divided them into categories depending on the adapted type of approach. I distinguished probabilistic, enrichment-based, regression-based, convex hull, matrix factorization and attractor metagene approaches that can be used for cell-type deconvolution. I detailed the basis of the different models and highlighted the most important features counting for cell-type deconvolution.

DNAm data were also used to estimate cell-type proportions. However, the heterogeneity found in methylome data resulting from the difference in cell type proportions is usually seen as a confounding factor to be removed. CNA data can be used for estimation of tumor purity and clonality.

In brief, for the transcriptome cell-type deconvolution, it can be observed that just a limited number of tools are usable in practice in order to deconvolute large cancer cohorts and without the need to provide hard to estimate parameters. Supervised methods applied to cancer use reference-profiles coming from a different context. Unsupervised tools, so far, are somewhat underrepresented in the field and do not offer a solution directly applicable to cancer transcriptomes of high dimensions. All of the presented methods are still waiting for consistent validation with the gold standard benchmark. This could be done if systematic data of bulk transcriptome paired with FACS-measured cell-type proportions information for many cells and in many samples were generated. Another unanswered question is the validity of the linear mixture model in the presence of cell-cell interactions.

There is still a room for improvement in the field in order to provide more user-friendly, accurate and precise cell-type abundance estimations.

A question can be asked, **are cell-type proportions enough to understand tumor immune phenotypes?** Can we extract more valuable information from the bulk omic data that would give useful insight to biological functions of the *in silico* dissected cells?

Objectives

In the introduction, I have described two sides of studying TME complexity. I placed in the context of cancer research and cancer therapy the most recent studies of tumor immunity with a focus on system-level computational approaches. I have also introduced a wide array of available approaches to address deconvolution of bulk omic data. I reviewed their strong and weak points, and I presented general trends since the field was established.

Answers to important questions on *how TME modulates tumor*, *how to propose better cancer subtyping for immune therapies* and *how to predict better response to treatment* are perhaps **hidden in already generated bulk omic data**. However, new methodological tools and a more overall view is needed to uncover hidden patterns better.

In this thesis, I aim to bring new insights into composition and function of TME. It is clear that complex information is necessary to understand the role of different immune cells in cancer and not only presence but also function are to be deciphered from available data. Therefore, this project, on its biological side, has two main aims:

1. fundamental research: understand the presence of different cell type, their interactions and functions in TME of different cancers types and how other factors as stress, cell cycle, etc. shape them. Thanks to data-driven and discovery nature of the project, I will also hope to understand how the signature of cell type evolves in different conditions shaped by other cells and factors.
2. translational research: how immune landscape and its state can help to predict patient survival and better tailor recommendation for therapy. The analysis could also bring to the light possible biomarkers or drug targets for immune therapies.

I aim to explore publicly available data, challenge inter-lab, and inter-platform biases. I will use mainly bulk transcriptomic data (because of available volume) and cross-validated with other data types: scRNA-seq, FACS, IHC when possible.

On its computational/mathematical side it will face following challenges:

1. Establish state-of-art of existing bulk deconvolution methods, discuss their advantages and limits

2. Propose new unsupervised method that will fill the knowledge gap giving an insight into context-specific signatures of cell types/cell states in cancer
3. Deliver well-documented and user-friendly tool that can be used by the scientific community
4. Decompose a big corpus of bulk omic data into interpretable biological functions, with a particular focus on the immune cell types
5. Use different data types (scRNAseq, microarray, RNAseq, FACS, etc.) to complete, compare and contrast findings of the analysis.
6. Decompose established immune cells populations from metastatic melanoma in order to better understand cell-type heterogeneity

In order to face these challenges, I have first focused on testing and creating new methods. This is why methods and results are interlaced in this thesis. Reproducing work of other researchers it is not always easy, and sometimes it is even impossible. Much time was invested in understanding and reusing previous publications, part of this effort was reflected in the introduction, some of my thoughts will be expressed in the discussion.

Next important step was improving and testing ideas born in our team. I collaborated to a publication on a topic Chapter 3, and I have authored an extension of this work described in Chapter 4. I have also compared my tool to other similar methods, an overview of the results are in Chapter 5.

Once I have found the most appropriate way to apply my method, that I validated with multiple datasets, I have built a tool to share it with the scientific community (Chapter 6). The tool is freely available online as an R package. During my work, I have collected many datasets of tumor signatures, tumor metagenes, benchmark datasets some of which are part of my tool. I have also accessed, thanks to the courtesy of our collaborators a collection of pan-cancer bulk transcriptomic datasets that I compared with other publicly available datasets. I build my working environment in which I managed and cleaned the data.

Finally, I realized a pan-cancer analysis of over 100 datasets which is the primary outcome of my work. I completed results of this work with published scRNAseq data from tumor samples. This analysis is a source of precious information, I have, so far, explored only part of possible direction focusing on cancer infiltrating T-cells. This results will be found in a manuscript in preparation in Chapter 7. However, more information can still be extracted in the further work. There is also a possibility to provide experimental validation of my findings, and it will be considered in the perspectives part.

In parallel, I used part of methods to study cell-type heterogeneity in an independent project resulted in a publication in review (Chapter 8).

The remaining time, I have invested in collaborations and contributions to different works within and outside of my team. Published work from those projects will be shortly described in Annexes.

Part II

Results

Chapter 3

Determining the optimal number of independent components for reproducible transcriptomic data analysis

Ulykbek Kairov*, Laura Cantini*, Alessandro Greco, Askhat Molkenov, Urszula Czerwinska, Emmanuel Barillot and Andrei Zinovyev

* contributed equally

Published in BMC Genomics, 11 September 2017

3.1 Context

In the introduction to the computational cell-type deconvolution, I have introduced Matrix factorization *family* of approaches including, Independent Components Analysis (ICA). ICA decomposes transcriptome (X) into to matrices of sources (S) and mixing matrix (A). I have mentioned that one of the most critical parameters to decompose transcriptome with matrix factorization methods is the parameter that I called k (called M through this publication), which is the number of sources (independent components).

3.2 Description

In this publication, [?] developed a way to identify a Maximally Stable Transcriptomic Dimension (MSTD). This index helps to decompose transcriptomes into interpretable biological factors. We mention that different methods to find the right number of k were developed in the previous works. However, none was conceived with biological interpretation clarity in mind. The MSTD index is computed based on the stability of the components over different k . The details of how MSTD is computed are explained in the publication.

The components coming from decompositions of within MSTD range have a higher probability to be found in other transcriptome datasets. In this way, reproducible signatures of cancer transcriptomes can be identified in diverse cancer datasets, published by different authors and in different platforms.

The concept was tested on TCGA data and six independent breast cancer datasets (37 datasets in total).

In overall, we observed that average stability of computed components decreases when the number of components increase (Fig 1. [?]), while the top components keep their stability almost unchanged.

If one uses $k < \text{MSTD}$, it makes the identification of biological signals difficult because different factors (sources) remain merged or mixed.

An unexpected observation while working on MSTD was that if the transcriptome is decomposed into a high number of $k \gg \text{MSTD}$, the signals existing in lower dimensions are robustly identifiable. Besides, an important observation was made in METABRIC dataset, where the *Immune* signal existing in MSTD dimension, in high k , splits into three signals identifiable as groups of immune cells (Fig. 2f [?]).

A side-effect of $k \gg \text{MSTD}$ were also components driven by a small number of genes. This phenomenon is described in details in the publication.

3.3 Impact on the further work

I have contributed to this publication by running numerous simulations and working on the final manuscript.

This work was a significant breakthrough in my work on cell-type deconvolution. While the co-authors started to develop the MSTD index, I was working on finding the best k for immune cells identification. I was trying mainly two different strategies:

1. decomposition of the transcriptome matrix with fastICA into a very stable dimension $k \approx 10$, selection of the immune signal, selection of the n top genes of these immune components from the transcriptome and running the fastICA again

2. trying different decompositions and interpreting results, with an objective to define best k for the interpretability.

I have presented the first strategy (1), that was giving promising results on a breast carcinoma dataset, at [ISMB conference in 2016](#). I gave a short talk and presented an [award-winning poster](#) presenting the strategy. However, this strategy was not easy to generalize and apply to multiple datasets.

Thus, I started to experiment with the second strategy. However, it was not easy to evaluate the quality of decomposition, as I was employing gene enrichment methods (like GSEA or Fisher test described in the previous chapter) that are not free from false positives.

Finally, participating in the work on this publication, I have found a third possibility:

3. decomposing transcriptome into a high number of components ($k \gg MSTD$) that allows direct identification of cell-type-specific components

The possibility to apply this strategy reproducibly remained unclear. The *unstable* components were not supposed to be found with high probability in other transcriptome data.

In the next chapter, I describe the study where I test the third strategy in multiple cancer transcriptome datasets.

RESEARCH ARTICLE

Open Access



Determining the optimal number of independent components for reproducible transcriptomic data analysis

Ulykbek Kairov^{2†}, Laura Cantini^{1†}, Alessandro Greco¹, Askhat Molkenov², Urszula Czerwinska¹, Emmanuel Barillot¹ and Andrei Zinovyev^{1*+ID}

Abstract

Background: Independent Component Analysis (ICA) is a method that models gene expression data as an action of a set of statistically independent hidden factors. The output of ICA depends on a fundamental parameter: the number of components (factors) to compute. The optimal choice of this parameter, related to determining the effective data dimension, remains an open question in the application of blind source separation techniques to transcriptomic data.

Results: Here we address the question of optimizing the number of statistically independent components in the analysis of transcriptomic data for reproducibility of the components in multiple runs of ICA (within the same or within varying effective dimensions) and in multiple independent datasets. To this end, we introduce ranking of independent components based on their stability in multiple ICA computation runs and define a distinguished number of components (Most Stable Transcriptome Dimension, MSTD) corresponding to the point of the qualitative change of the stability profile. Based on a large body of data, we demonstrate that a sufficient number of dimensions is required for biological interpretability of the ICA decomposition and that the most stable components with ranks below MSTD have more chances to be reproduced in independent studies compared to the less stable ones. At the same time, we show that a transcriptomics dataset can be reduced to a relatively high number of dimensions without losing the interpretability of ICA, even though higher dimensions give rise to components driven by small gene sets.

Conclusions: We suggest a protocol of ICA application to transcriptomics data with a possibility of prioritizing components with respect to their reproducibility that strengthens the biological interpretation. Computing too few components (much less than MSTD) is not optimal for interpretability of the results. The components ranked within MSTD range have more chances to be reproduced in independent studies.

Keywords: Transcriptome, Independent component analysis, Reproducibility, Cancer

Background

Independent Component Analysis (ICA) is a matrix factorization method for data dimension reduction [1]. ICA defines a new coordinate system in the multi-dimensional space such that the distributions of the data point projections on the new axes become as mutually independent as possible. To achieve this, the standard approach is maximizing the non-gaussianity of the data

point projection distributions [1]. ICA has been widely applied for the analysis of transcriptomic data for blind separation of biological, environmental and technical factors affecting gene expression [2–6].

The interpretation of the results of any matrix factorization-based method applied to transcriptomics data is done by the analysis of the resulting pairs of metagenes and metasamples, associated to each component and represented by sets of weights for all genes and all samples, respectively [6, 7]. Standard statistical tests applied to these vectors can then relate a component to a reference gene set (e.g., cell cycle genes), or to clinical

* Correspondence: Andrei.Zinovyev@curie.fr

†Equal contributors

¹Institut Curie, PSL Research University, INSERM U900, Mines ParisTech, Paris, France

Full list of author information is available at the end of the article

annotations accompanying the transcriptomic study (e.g., tumor grade). The application of ICA to multiple expression datasets has been shown to uncover insightful knowledge about cancer biology [3, 8]. In [3] a large multi-cancer ICA-based metaanalysis of transcriptomic data defined a set of metagenes associated with factors that are universal for many cancer types. Metagenes associated with cell cycle, inflammation, mitochondria function, GC-content, gender, basal-like cancer types reflected the intrinsic cancer cell properties. ICA was also able to unravel the organization of tumor microenvironment such as the presence of lymphocytes B and T, myofibroblasts, adipose tissue, smooth muscle cells and interferon signaling. This analysis shed light on the principles underlying bladder cancer molecular subtyping [3].

It has been demonstrated that ICA has advantages over the classical Principal Component Analysis (PCA) with respect to interpretability of the resulting components. The ICA components might reflect both biological factors (such as proliferation or presence of different cell types in the tumoral microenvironment) or technical factors (such as batch effects or GC-content) affecting gene expression [3, 5]. However, unlike principal components, the independent components are only defined as local minima of a non-quadratic optimization function. Therefore, computing ICA from different initial approximations can result in different problem solutions. Moreover, in contrast to PCA, the components of ICA cannot be naturally ordered.

To improve these aspects, several ideas have been employed. For example, an *icasso* method has been developed to improve the stability of the independent components by: (1) applying multiple runs of ICA with different initializations; (2) clustering the resulting components; (3) defining the final result as cluster centroids; and (4) estimating the compactness of the clusters [9]. The resulting components can be then naturally ordered from the most stable to the least stable ones. This ranking is usually different from more commonly used independent component rankings based on the value of the used non-gaussianity measure (such as kurtosis) or the variance explained by the components.

The fundamental question is the determination of the number of independent components to produce. This problem can be split into two parts: a) what dimension should be selected for reducing the transcriptomic data before applying ICA (determining the effective data dimension); and b) which is the most informative number of components to use in the downstream analysis?

Determining the optimal effective data dimension for application of signal deconvolution was a subject of research in various fields. For example, ICA appeared to be a powerful method for analyzing the fMRI (functional magnetic resonance) data [9–12]. In this field, it was

shown that choosing a too small effective data dimension might generate “fused components,” not reflecting the heterogeneity of the data, leading to a loss of interesting sources (under-decomposition). At the same time, choosing the effective dimension too high might lead to signal-to-noise ratio deterioration, overfitting and splitting of the meaningful components (over-decomposition) [10–12]. The influence of the effective dimension choice on the ICA performance has not been well studied in the context of transcriptomic data analysis. For example, in [3] each dataset was decomposed into a number of components in an ad hoc manner ($n = 20$).

Several theoretical approaches for estimating effective data dimension exist. The simplest ones, developed for PCA analysis, are represented by the Kaiser rule aimed at keeping a certain percentage of explained variance and the broken stick model of resource distribution [13]. More sophisticated approaches employ the information theory (e.g., Akaike’s information or Minimal Description Length criteria) [13] or investigate the local-to-global data structure organization [14]. Also, computational approaches based on cross-validation have been suggested in the literature [15]. Specifically for ICA analysis, few methods have been proposed to optimize the effective dimension. For example, the Bayesian Information Criterion (BIC) can be applied to the Bayesian formulation of ICA for selecting the optimal number of components [16].

Although many of the above theoretical methods are “parameter-free,” selecting the best method for choosing an effective dimension for transcriptomic data can be challenging in the absence of a clearly defined validation strategy. One possible approach to overcome this limitation is to apply the same computational method to multiple transcriptomic datasets derived from the same tissue and disease. In this situation, it is reasonable to expect that a matrix factorization method should detect similar signals in all datasets. By taking advantage of the rich collection of public data such as The Cancer Genomic Atlas (TCGA) [17] and Gene Expression Omnibus [18], it is possible to compare and contrast the parameters of different gene expression analysis methods such as ICA.

In this study, we used TCGA pan-cancer (32 different cancer types) transcriptomic datasets and a set of six independent breast cancer transcriptomic datasets to evaluate the effect of the number of computed independent components on reproducibility and biological interpretability of the obtained results. We evaluated the reproducibility of ICA on three aspects: First, we analyzed the stability of the computed components with respect to multiple runs of ICA; second, we analyse the conservation of the computed components by varying the choice of the reduced data dimension; and third, we consider the reproducibility of the resulting set of ICA

metagenes across multiple independent datasets. Our reproducibility analysis thus explores 13,027 transcriptomic profiles in 37 transcriptomic datasets, for which more than 100,000 ICA decompositions have been computed.

We finally defined a novel criterion adapted for choosing the effective data dimension for ICA analysis of gene expression, which takes into account the global properties of transcriptomic multivariate data. The Maximally Stable Transcriptome Dimension (MSTD) is defined as the maximal dimension where ICA does not yet produce a large proportion of highly unstable signals. By numerical experiments, we showed that components ranked by stability within the MSTD range tend to be more reproducible and easier to interpret than higher-order components.

Results

Definition of component reproducibility measures used in this study

Stability of an independent component, in terms of varying the initial starts of the ICA algorithm, is a measure of internal compactness of a cluster of matched independent components produced in multiple ICA runs *for the same dataset and with the same parameter set but with random initialization*. The exact index used for quantifying the clustering is documented in the Methods section. Conservation of an independent component in terms of choosing various orders of ICA decomposition is a correlation between matched components computed in two ICA decompositions of different orders (reduced data dimensions) *for the same dataset*. Reproducibility of an independent component is an (average) correlation between the components that can be matched after applying the ICA method using the same parameter set but *for different datasets*. For example, if a component is reproduced between the datasets of the same cancer type, then it can be considered a reliable signal less affected by technical dataset peculiarities. If the component is reproduced in datasets from many cancer types, then it can be assumed to represent a universal carcinogenesis mechanism, such as cell cycle or infiltration by immune cells. The details on computing correlations between components from different datasets are described in Methods.

Maximally stable Transcriptome dimension (MSTD), a novel criterion for choosing the optimal number of ICs in transcriptomic data analysis

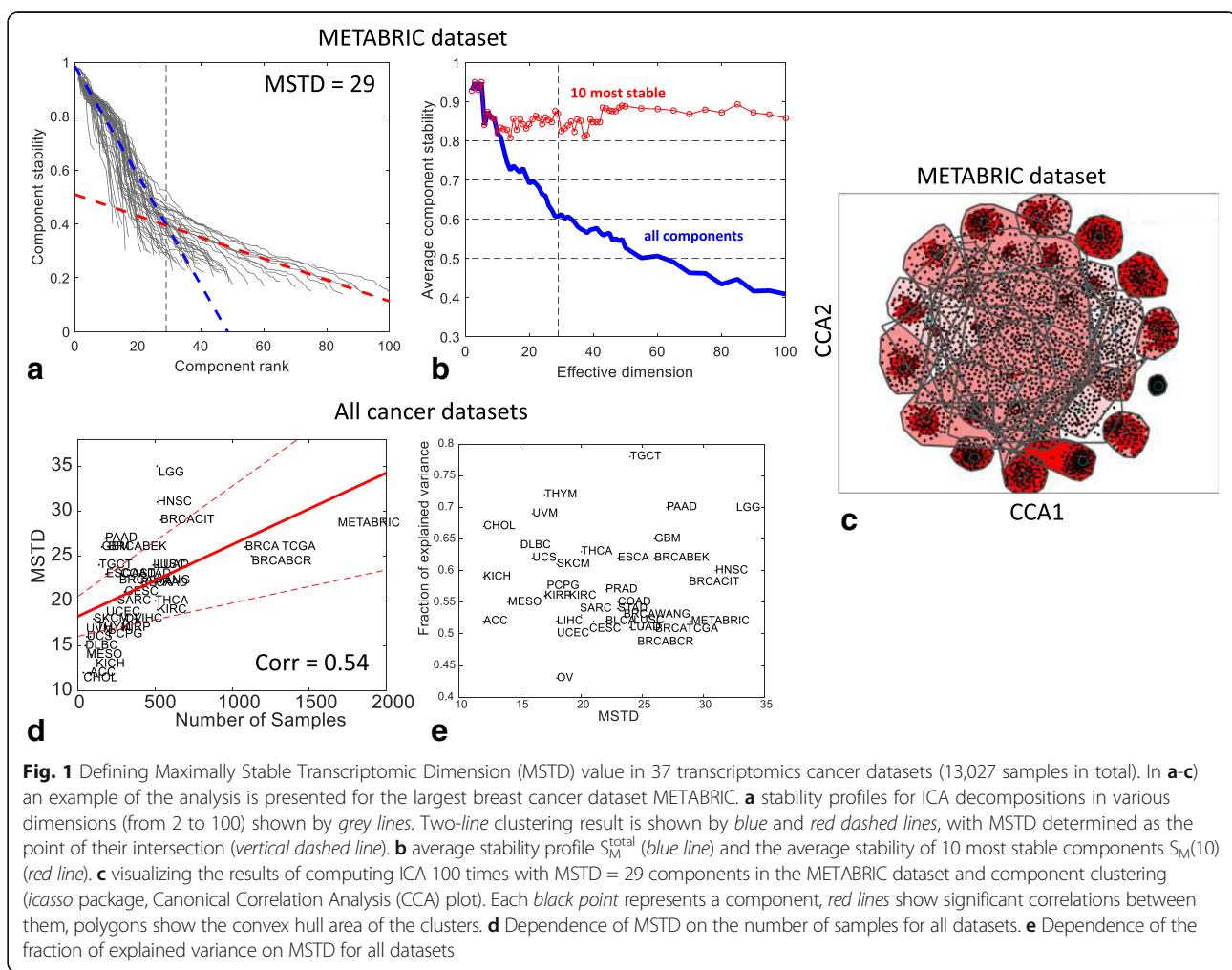
We used 37 transcriptomic datasets to analyze the stability and reproducibility of the ICA results conditional on the chosen number of components. ICA has been applied separately to 37 cancer transcriptomic datasets

following the ICA application protocols as described in Methods.

The proposed protocol depends on a fundamental parameter M (effective dimension of the data and, at the same time, the number of computed independent components) whose effect on the stability of the ICs is investigated. For each transcriptomic dataset, the range of M values 2–100 has been considered. For each value of M , the data dimension is reduced to M by PCA and then data whitening is applied. Subsequently, the actual signal decomposition is applied in the whitened space by defining M new axes, each maximizing the non-gaussianity of data point projections distribution.

For transcriptomic data, ICA decomposition provides: (a) M metagenes ranked accordingly to their stability in multiple runs ($n = 100$) of ICA; and (b) a profile of stability of the components (set of M numbers in [0,1] range in descending order). Considering the largest dataset METABRIC as an example, the behavior of the stability profile as a function of M is reported in Fig. 1a. The results for stability analysis for other breast cancer datasets are similar (See Additional file 1: Figure SF2). To recapitulate the behaviour of many stability profiles, the average stability of the first k top-ranked components $S_M(k)$ is used (See Fig. 1b). For $k = M$, the average stability of all computed components is denoted as S_M^{total} . Three major conclusions can be made from Fig. 1. First, the average stability of the computed components S_M^{total} decreases with the increase of M , while the average stability of the first few top ranked components, e.g., $S_M(10)$, weakly depends on M (Fig. 1b). Moreover, S_M^{total} is characterized by the presence of local maxima, defining certain distinguished values of M that correspond to the (locally) maximally stable set of components (Fig. 1b). Third, the stability profiles for various values of M can be classified into those for which the stability values are distributed approximately uniformly and those (usually, in higher dimensions) forming a large proportion of the components with low stability (I_q between 0.2 and 0.4) (Fig. 1a).

Considering these observations, we hypothesized that the optimal number of independent components – large enough to avoid fusing meaningful components and yet small enough to avoid producing an excessive amount of highly unstable components – should correspond to the inflection point in the distribution of the stability profiles (Fig. 1a). To find this point, the stability measures have been clustered along two lines, which is analogous of 2-means clustering but with lines as centroids. In this clustering, the line with a steeper slope (Fig. 1a, blue line) grouped the stability profiles with uniform distribution, while another line (Fig. 1a, red line) matched the mode of low stability components. The intersection of these lines provided a consistent estimate of the effective



number of independent components. We call this estimate Maximally Stable Transcriptome Dimension (MSTD) and in the following we investigated its properties. We note that, as in various information theory-based criteria (BIC, AIC), this estimate is free of parameters (thresholds), and it only exploits the property of the qualitative change in the character of the stability profile in higher data dimensions for transcriptomic data.

In most of the cancer transcriptomics datasets used in our analysis, MSTD was found to correspond roughly to the average stability profile $S_M^{\text{total}} \approx 0.6$ (Additional file 1: Figure SF2). In Fig. 1d, the dependence of MSTD on the number of samples contained in the transcriptomic dataset is investigated for all the 37 transcriptomic datasets. As shown in Additional file 2: Figure SF1, MSTD increased with the number of samples; however, this trend was weaker than other estimates of an effective dimension such as Kaiser rule and broken stick distribution-based data dimension estimates. Finally, the fraction of variance explained by the linear subspace spanned by MSTD number of components was evaluated (Fig. 1e),

and it was observed that the fraction of variance explained varied from 0.45 to 0.75 with a median of 0.56.

Underestimating the effective dimension ($M < \text{MSTD}$) leads to a poor detection of known biological signals

Previous large-scale ICA-based meta-analyses [3] have shown that some of the ICs derived from the decomposition of a cancer transcriptomic data were clearly and uniquely associated with known biological signals. For example, one of these signals was the one connected to proliferative status of tumors. Another example was given by the signals related to the infiltration of immune cells that were also strongly heterogeneous across cancer patients.

We have checked the reproducibility of several metagenes obtained in previous meta-analyses [3] for all ICA decompositions as a function of M . For this analysis, we employed the METABRIC breast cancer dataset, which was not included in the input data of the previous publication [3] and thus it had not been used to derive the metagenes of that work. In addition, we checked how

the significance of intersections between the genes defining the components and several reference gene sets (produced independently of the ICA analyses) behaved as a function of M .

We applied the previously developed correlation-based approach to match previously identified metagenes with the ones computed for a new METABRIC dataset (see Methods section). The components were oriented accordingly to the direction of the heaviest tail of the projection distribution. When matching an oriented component to the previously defined set of metagenes, we verified that the resulting maximal correlation should be positive, i.e. large positive weights in one metagene should correspond to large positive weights in another metagene.

One of the most important case studies is reproducibility of the “proliferative” metagene in different data dimensions. It is investigated in Fig. 2a-c. For this metagene, we computed correlations with M newly identified independent components. As an example, the profile of correlations for $M = 100$ is shown in Fig. 2b. It can be seen that one of the components (ranked #7 by stability analysis) is much better correlated to the proliferative metagene than any other component. Therefore, component #7 is called “best matched” in this case, for $M = 100$, and “well separable.” Repeating this analysis for all M and reporting the observed maximal correlation coefficient and the corresponding stability value gives a plot shown in Fig. 2a. Separability of the best matched component from the other components is visualized in Fig. 2c.

As it can be seen from Fig. 1a, the biologically expected signals (i.e., cell cycle) can be poorly detected for $M < \text{MSTD}$; however, once the best matching component with significant correlation was found, it remained unique and was detected robustly even for very large values of $M > \text{MSTD}$. For example, even when 100 components (M) were computed, the correlation between the previously defined proliferative metagene and the best matched independent component did not diminish (Fig. 2a). Moreover, the separability of the best matched component from the rest of the components was not ruined (Fig. 2c). In this example, the identification of cell cycle component remained clear (large and well-separated correlation coefficient) for $M > \text{MSTD}$. This result was consistent and complementary when compared with the previously observed weak dependence of $S_M(10)$ on M . Indeed, the “proliferative” best matched component had stability rank k in the range [6, 11]. That is, it remained stable in ICA decompositions in all dimensions. Moreover, the intersection of a recently established proliferation gene signature [19] with the set of top contributing genes of the best matched component improved with increasing M and saturated (Fig. 2d). This proves that the detection of the proliferation-associated signal with

ICA does not depend on the ICA-based definition of the proliferative metagene.

Together with the proliferative signal, other metagenes from the previously cited ICA-based meta-analysis [3] were robustly identified in our analysis. In Fig. 2e-h, we showed the correlation with the best matching component for the metagenes associated with the presence of myofibroblasts, inflammation, interferon signaling and immune system, as a function of M . These plots illustrated different scenarios that can result from such analysis. The myofibroblast-associated metagene was robustly detected for all values of $M > 7$ (Fig. 2f). However, the stability of the best matching component was deteriorated in higher-order ICA decompositions ($M > 45$). For the inflammation-associated metagene, an ICA decomposition with $M > 38$ was needed to robustly detect a component that correlates with the metagene (Fig. 2e).

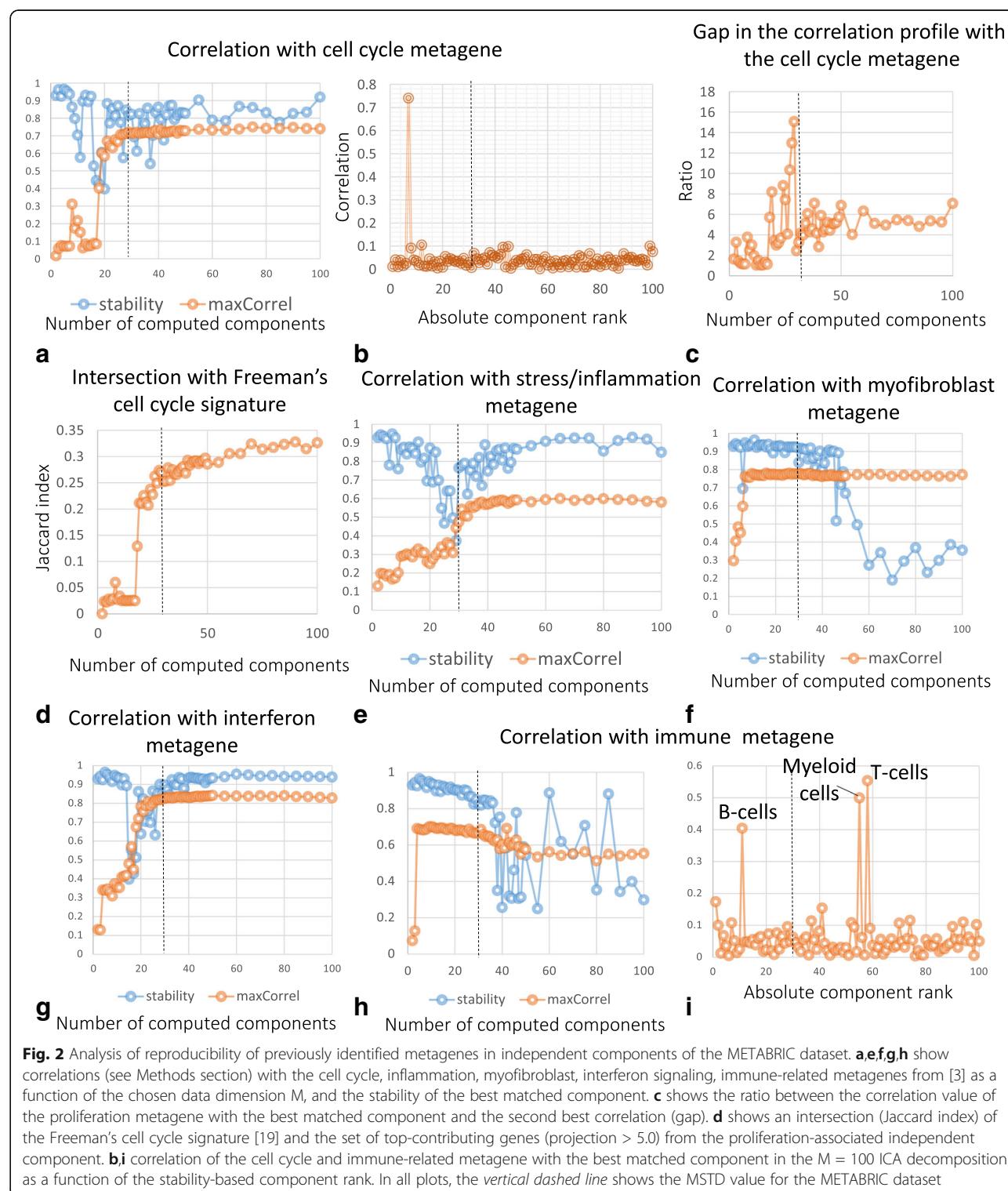
Interestingly, the immune-associated metagene was found robustly matched starting from $M = 4$. However, in higher-order decompositions (starting from $M = 30$) it could be matched to several components that can be associated with specific immune system-related signals (Fig. 2h-i). Hypergeometric tests applied to the sets of top-contributing genes (weights larger than 5.0) allowed us to reliably interpret these components as being associated with the presence of three types of immune-related cells: T cells (corrected enrichment p -value = 10^{-39} with “alpha beta T cells” signature [20], other immune signatures are much less significant), B cells (p -value = 10^{-7} with “B cells, preB.FrD.BM” signature) and myeloid cells (p -value = 10^{-78} with “Myeloid Cells, DC.11cloSer.Salm3.SI” signature).

Overestimating the number of components ($M > \text{MSTD}$) produces multiple ICs driven by small gene sets

We observed that the higher-order ICA decompositions ($M > \text{MSTD}$) produced a larger number of components driven by small gene sets (frequently, one gene), such that the projections of the genes in this “outlier” set is separated by a relatively large gap with the rest of the projections. We thus designed a simple algorithm to distinguish such components driven by a small gene set from all the others. The names of the genes composing these small sets were used for annotating the corresponding components (Fig. 3a, right part).

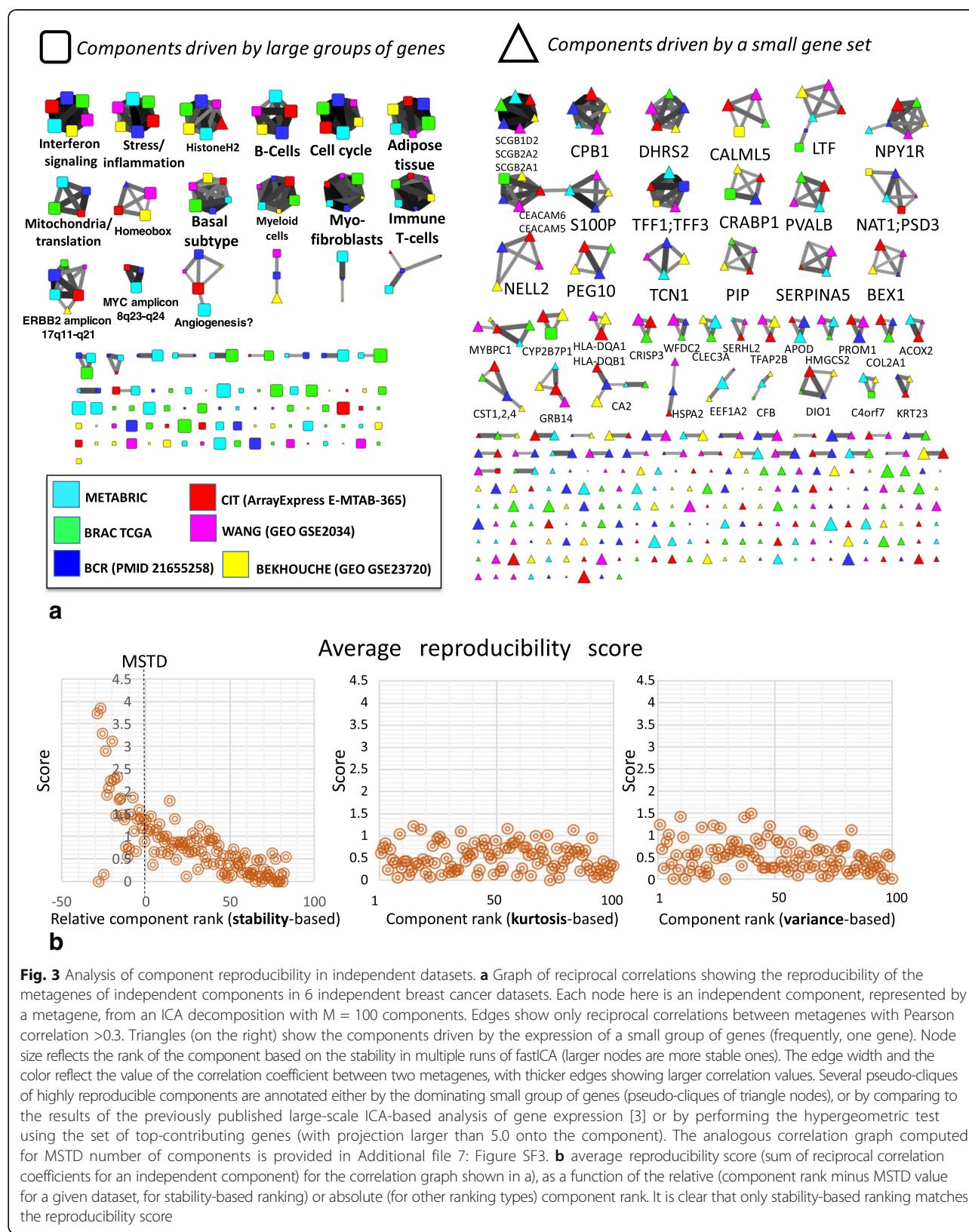
It was observed that the presence of such “small gene set-driven” components is a characteristic of higher-order ICA decompositions ($M > \text{MSTD}$), much less present in ICA decompositions with $M \leq \text{MSTD}$ (compare Fig. 3a and Additional file 1: Figure SF2).

To check the biological significance of the outlier genes, we considered as a case study the higher-order ($M = 100$) ICA decomposition of the METABRIC breast cancer dataset. We collected all those genes found to be



drivers of at least one “small gene set-driven” component. We obtained in this way a set of 98 genes listed in Additional file 3: Table ST2. This list appeared to be strongly enriched ($p\text{-value} = 10^{-12}$ after correction for multiple testing) in the genes of the signature

DOANE_BREAST_CANCER_ESR1_UP “Genes up-regulated in breast cancer samples positive for ESR1 compared to the ESR1 negative tumors” from Molecular Signature Database [21] and several other specific to breast cancer gene signatures. This analysis thus



suggested that at least some of the identified “small gene set-driven” components are not the artifacts of the ICA decomposition, but they can be biologically meaningful and reproducible in independent datasets (Fig. 3a, right part).

Most stable components with stability rank \leq MSTD have more chances to be reproduced across independent datasets for the same cancer type

It would be reasonable to expect that the main biological signals characteristic for a given cancer type should be the same when one studies molecular profiles of different independent cohorts of patients. Therefore, we expect that for multiple datasets related to the same cancer type, ICA decompositions should be somewhat similar; hence, reciprocally matching each other. We called this expected behavior “reproducibility,” and here we studied this by applying ICA to six relatively large breast cancer transcriptomic datasets. Of note, these datasets were produced using various technologies of transcriptomic profiling (Additional file 4: Table ST1).

To identify the reproducible components, we applied the same methodology as in the previously published ICA-based gene expression meta-analysis [3]. We decomposed the six datasets separately and then constructed a graph of reciprocal correlations between the obtained metagenes. Correlation between two sets of components is called reciprocal when a component from one set is the best match (maximally correlated) to a component from another set, and vice versa (see Methods for a strict definition).

Pseudo-cliques in this graph, consisting of several nodes, correspond to reproducible signals detected by ICA. As shown in Fig. 3, multiple reproducible signals were identified in the analysis. Some of them correspond to signals already identified in [3] (e.g., cell cycle, interferon signaling, microenvironment-related signals), and some correspond to newly discovered biological signals (e.g., ERBB2 amplicon-associated). Some other pseudo-cliques are associated with “small gene set-driven” components (frequently, one gene-driven), such as TFF1–3-associated or SCGB2A1–2-associated components.

The genes driver of reproducible and “small gene set-driven” components (S100P, TFF1, TFF3, SCGB2A1, SCGB1D2, SCGB2A2, LTF, CEACAM6, CEACAM5 being most remarkable examples) have been investigated in detail, to further check their biological interest. They were found to be the genes known to be associated with breast cancer progression [22]. For example, seven of the nine previously mentioned genes form a part of a gene set known to be up-regulated in the bone relapses of breast cancer (M3238 gene set from MSigDB).

To quantify the reproducibility of the components, we computed a reproducibility score. It is a sum of

correlation coefficients between the component and all reciprocally correlated components from other datasets. By construction, the maximum value of the score is 5, which meant that a component with such a score would be perfectly correlated with the reciprocally related components from five other datasets. We studied the dependence of this score as a function of the relative to MSTD component stability-based rank (Fig. 3b). From this study, it follows that even for the high-order ICA decompositions, the components ranked by their stability within MSTD range, have an increased likelihood of being reproduced in independent datasets collected for the same cancer type.

To show that the stability-based ranking of genes is more informative compared with the standard rankings of independent components, we performed a computational analysis in which we compared the stability-based ranking with the rankings based on non-gaussianity (kurtosis) and explained variance. These two measures are frequently used to rank the independent components [6]. From Fig. 3b it is clear that the stability-based ranking of independent components corresponds well to the reproducibility score, while two other simpler measures do not.

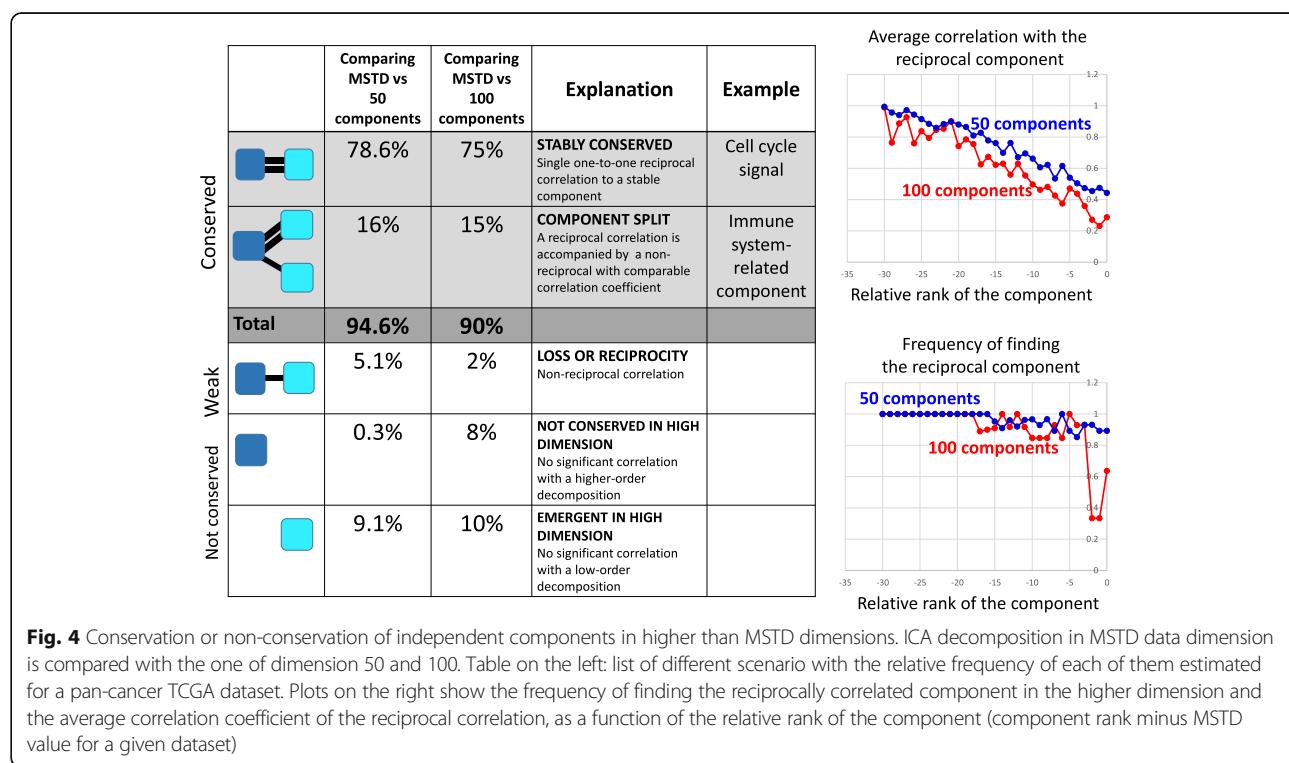
It can also be shown that the total number of reciprocal correlations with relatively large correlation coefficients ($|r| > 0.3$) between ICA-based metagenes computed for several independent datasets is significantly bigger when the component stabilization approach is applied (Additional file 5: Figure SF4). This proves the utility of the applied stabilization-based protocol of ICA application to transcriptomic data.

Computing large number of components ($M > MSTD$) does not strongly affect the most stable ones

We lastly used ICA decompositions of 37 transcriptomic datasets to compare the ICA decompositions corresponding to $M = MSTD$ with the higher-order decompositions, $M = 50$ or $M = 100$.

It was found that the components calculated in lower data dimensions can be relatively well matched to the components from higher-order ICA decompositions (Fig. 4). More precisely, 90% of the components defined for $M = MSTD$ had a reciprocal best matched component in the $M = 100$ ICA decomposition. Most stable components had a clear tendency to be reproduced with high correlation coefficient ($r > 0.8$). Only 10% of the components had only non-reciprocal or too small correlations between two decompositions (in other words, *not conserved* in higher-order ICA decompositions).

Approximately 15% of the components in $M = MSTD$ ICA decomposition together with reciprocal maximal correlation also had a non-reciprocal correlation to one of the components in $M = 100$ ICA decomposition (Fig. 4). This case can be described as splitting a component into



two or more components in the higher-order ICA decompositions. At least one such split had a clear biological meaning, namely the splitting of the component representing the generic “immune infiltrate.” The resulting “split” components more specifically represented the role of T cells, B cells and myeloid cells in the tumoral micro-environment (see the “*Underestimating the effective dimension...*” Results section).

Discussion

Our results shed light on the organization of the multivariate distribution of gene expression in the high-dimensional space. It appears that the organization contained two relatively well separated parts: *the dense one* of a relatively small effective dimension and *the sparse one*. The former contained the genes from within co-regulated modules that contained from few tens to few hundreds of genes. The latter was spanned by the genes with unique regulatory programs (perhaps tissue-specific) weakly shared by the other genes. Here the sparsity was understood in the sense of low local multivariate distribution density.

Independent Component Analysis can capture both these parts of the multivariate distribution. However, while the dense part defined independent components with approximately uniformly distributed stabilities, starting from highly stable to less stable, the sparse part was spanned by the components characterized mostly by small stability values.

This organization of the gene expression space is captured in the distribution of ICA stability profiles for varying M , which allowed us to define the Maximally Stable Transcriptome Dimension (MSTD) value, roughly reflecting the dimension of the dense part of the gene expression distribution. In one hand, when underdecomposing (compressing too much by dimension reduction, $M < \text{MSTD}$) a transcriptomic dataset, the resulting independent components are hard to interpret. In the other hand, overdecomposing transcriptomes (choosing the effective dimension much bigger than MSTD) is not dramatically detrimental: one can choose to explore a relatively multi-dimensional subspace of a transcriptomic dataset, taking into account that applying matrix factorization methods in higher dimensions becomes computationally challenging and prone to bad algorithm convergence. Nevertheless, higher-order decompositions might allow capturing the behavior of some tissue-specific or cancer type-specific biomarker genes from the sparse part of the distribution, which can be found reproducible in other independent studies.

In our computational experiments, we selected 100 as the maximum order of ICA decomposition (M) to test. However it is possible to examine even higher orders of ICA decompositions, reducing the data to more than 100 dimensions, but not more than the total number of samples, of course. In practice, computing ICA in such high dimension leads to significant deterioration of the fastICA algorithm convergence, so exploring $M > 100$

might be too expensive in terms of computational time. Moreover, our study suggests that the most interesting for interpretation components are usually positioned within the first few ten top ranks: therefore, 100 seems to be a reasonable limit for dimension reduction when applying ICA to transcriptomic data.

Our proposed approach can be used for comparing intrinsic reproducibility, at different levels, of various matrix factorization methods. For example, it would be of interest to compare the widely used Non-negative matrix factorization (NMF) method [6, 7] with ICA to assess reproducibility of extracted metagenes in independent datasets of the same nature.

More generally, systematic reproducibility analysis can be a useful approach for establishing the best practices of application of the bioinformatics methods.

Conclusion

By using a large body of data and comparing 0.1 million decompositions of transcriptomic datasets into the sets of independent components, we have checked systematically the resulting metagenes for their reproducibility in several runs of ICA computation (measuring *stability*), for their reproducibility between a lower order and higher-order ICA decompositions (*conservation*), and between metagene sets computed for several independent datasets, profiling tumoral samples of the same cancer type (*reproducibility*).

From the first of such analyses, we formulated a minimally advised number of dimensions to which a transcriptomic dataset should be reduced called Maximally Stable Transcriptome Dimension (MSTD). Reducing a transcriptomic dataset to a dimension below MSTD is not optimal in terms of the interpretability of the resulting ICA components. We showed that for relatively large transcriptomic datasets, MSTD could vary from 15 to 30 and that the number of samples matters relatively weakly.

From the second analysis, we concluded that the suggested protocol of ICA application to transcriptomic data is conservative, i.e., the components identified in a higher dimension (for example, in one hundred dimensional space) can be robustly matched with those components obtained in the dimensions comparable with MSTD. Moreover, we described an effect of interpretable component splitting in higher dimensions, leading to detection of finer-grained signals (e.g., related to the decomposition of the immune infiltrate in the tumor microenvironment). At the same time, the application of ICA in high dimensions resulted in a greater proportion of unstable components, many of them were driven by expression of small (one to three members) gene sets. Yet, some of these small gene set-driven components were highly reproducible and biologically meaningful.

From the third analysis, we established that the used protocol of ICA application, with ranking the independent components based on their stability, prioritized those components having more chances to be reproduced in independent transcriptomic datasets. Moreover, when ICA was applied in higher dimensions, the components within the MSTD range still have more chances to be reproduced.

In sum, our results confirmed advantageous features of ICA applied to gene expression data from different platforms, leading to interpretable and quantifiably reproducible results. Comparing ICA analyses performed in various dimensions and multiple independent datasets for the same cancer types allow prioritizing of the most reliable and reproducible components which can be quantitatively recapitulated in the form of metagenes or the sets of top contributing genes. We expect that ICA will demonstrate similar properties in other large-scale transcriptomic data collections such as scRNA-seq data.

Methods

Transcriptomics cancer data used in the analysis

Expression data derived for 32 solid cancer types (ACC, BLCA, BRCA, CESC, CHOL, COAD, DLBC, ESCA, GBM, HNSC, KICH, KIRC, KIRP, LGG, LIHC, LUAD, LUSC, MESO, OV, PAAD, PCPG, PRAD, READ, SARC, SKCM, STAD, TGCT, THCA, THYM, UCEC, UCS, UVM) were downloaded from the TCGA web-site and internally normalized. Normalized breast cancer datasets from CIT, BCR, WANG, BEKHOUCHE were re-used from the previous study [3]. Normalized METABRIC breast cancer expression dataset was downloaded from cBioPortal at this link http://www.cbioportal.org/study?id=brca_metabric. When it was not already the case, the data values were converted into logarithmic scale.

The list of breast cancer transcriptomic datasets used for reproducibility study is available in Additional file 4: Table ST1.

ICA decompositions computation

We applied the same protocol of application of ICA decomposition as in [3]. In the ICA decomposition $X \approx AS$, X is the gene expression (sample vs gene) matrix, A is the (sample vs. component) matrix describing the loadings of the independent components, and S is the (component vs. gene matrix) describing the weights (projections) of the genes in the components. To compute ICA, we used the *fastICA* algorithm [1] accompanied by the *icasso* package [23] to improve the components estimation and to rank the components based on their stability. ICA was applied to each transcriptomic dataset separately.

For each analysed transcriptomic dataset, we computed M independent components (ICs), using *pow3* nonlinearity and *symmetrical* approach to the decomposition, where $M = [2\dots 50, 55, 60, 65, 70, 75, 80, 85, 90, 95, 100]$. In those

cases, when M exceeded the total number of samples, the maximum M was chosen equal to 0.9 multiplied by the number of samples (moderate dimension reduction improves convergence). We found that the MATLAB implementations of *fastICA* performs superior to other implementations (such as those provided in *R* [24]). The computational time required for performing all the 0.1 million ICA decompositions used in this study is estimated in ~1500 single processor hours using MATLAB while other implementations would not make this analysis feasible at all. In our analysis, we used Docker with packaged compiled MATLAB code for *fastICA* together with MATLAB Runtime environment, which can be readily used in other applications and does not require MATLAB installed [25]. An example of computational time needed for the analysis of two transcriptomic datasets of typical size (full transcriptome, from 200 to 1000 samples) is provided in Additional file 6: Figure SF5. As a rough estimate, it takes 3 h to analyze a transcriptomic dataset with 200 samples and 7 h to analyze a dataset with 1000 samples, using an ordinary laptop. In each such analysis, more than 2000 ICA decompositions of different orders have been made.

The algorithm for determining the most stable Transcriptome dimension (MSTD)

- 1) Define two numbers $[M_{min}, M_{max}]$ as the minimal and maximal possible numbers of the computed components.
- 2) Define the number K of ICA runs for estimating the components stability. In all our examples, we used $K = 100$.
- 3) For each M between M_{min} and M_{max} (or, with some step) do
 - 3.1) Compute K times the decomposition of the studied dataset into M independent components using the *fastICA* algorithm. This results in computation of $M \times K$ components.
 - 3.2) Cluster $M \times K$ components into M clusters using agglomerative hierarchical clustering algorithm with the measure of dissimilarity equal to $1 - |r_{ij}|$, where r_{ij} is the Pearson correlation coefficient computed between components.
 - 3.3) For each cluster C_k out of M clusters (C_1, C_2, \dots, C_N) compute the stability index using the following formula

$$I_q(C_k) = \frac{1}{|C_k|^2} \sum_{i,j \in C_k} |r_{ij}| - \frac{1}{|C_k| \sum_{l \neq k} |C_l|} \sum_{i \in C_k} \sum_{j \in C_k} |r_{ij}|$$

where $|C_k|$ denotes the size of the k th cluster.

3.4) Compute the average stability index for M clusters:

$$S(M) = \frac{1}{M} \sum_k I_q(C_k)$$

- 4) Select the MSTD as the point of intersection of the two lines approximating the distribution of stability profiles (Fig. 1a). The lines are computed using a simple k-lines clustering algorithm [26] for $k = 2$, implemented by the authors in MATLAB, with the initial approximations of the lines matching the abscissa and the ordinate axes of the plot. The index used in 3.3 is a widely used index of clustering quality defined as a difference between the average intra-cluster similarity and the average inter-cluster similarity. In [9] this index was introduced to estimate the quality of clustering of independent components after multiple runs with random initial conditions, and tested in application to fMRI data. In the case of clustering independent components, $I_q = 1$ corresponds to the case of perfect clustering of components such that all the components in one cluster are correlated with each other with $|r| = 1$, and that all components in the same cluster are orthogonal to any other component (in the reduced and whitened space).

Comparing metagenes computed for different datasets and in different analyses

Following the methodology developed previously in [3], the metagenes computed in two independent datasets were compared by computing a Pearson correlation coefficient between their corresponding gene weights. Since each dataset can contain a different set of genes, the correlation is computed on the genes which are common for a pair of datasets. Note that this common set of genes can be different for different pairs of datasets. The same correlation-based comparison was done with previously defined and annotated metagenes. We computed the correlation only between those genes having projection value more than 3 standard deviations in the identified component.

When comparing two sets of metagenes $\mathbf{A} = \{A_1, \dots, A_M\}$ and $\mathbf{B} = \{B_1, \dots, B_N\}$, in order to do component matching, we focused on the maximal correlation of a metagene from one set with all components from another set. If $B_i = \arg \max(\text{corr}(A_j, \mathbf{B}))$ then B_i is called *best matched*, for A_j , metagene from the set \mathbf{B} . If $B_i = \arg \max(\text{corr}(A_j, \mathbf{B}))$ and $A_j = \arg \max(\text{corr}(B_i, \mathbf{A}))$, then the correlation between B_i and A_j is called *reciprocal*.

In all correlation-based comparisons, the absolute value of the correlation coefficient was used.

The orientation of independent components was chosen such that the longest tail of the data projection

distribution would be on the positive side. Then, for quantifying an intersection between a metagene and a reference set of genes (e.g., cell cycle genes), simple Jaccard index was computed between the reference gene set and the set of top-contributing genes to the component, with positive weights >5.0.

Determining if a small gene set is driving an independent component

To distinguish whether an independent component is driven by a small gene set, the distribution of gene weights W_i from the component was analyzed. For each tail of the distribution (positive and negative), the tail weight was determined as the total absolute sum of weights of the genes exceeding certain threshold W^{top} . The heaviest tail of the distribution was identified as the tail with the maximum weight. For the heaviest tail and for the set of genes P with absolute weights exceeding W^{top} , sorted in descending order by absolute value, we studied the gap distribution of values $G_i = W_i/W_{i+1}$, $i \in P$. If there was a single value of G_i exceeding a threshold G^{\max} , then the component was classified as being driven by a small set of genes corresponding to the indices $\{i; i \leq \max(k; G_k \leq G^{\max})\}$. The values $W^{\text{top}} = 3.0$, $G^{\max} = 1.5$ collected the maximal gene set size = 3 in all ICA decompositions. These are few genes with atypically high weights separated by a significant gap from the rest of the distribution (note that these genes cannot always be considered outliers since they and the resulting independent components can be reproducible in independent datasets).

Additional files

Additional file 1: Figure SF2. Estimating MSTD dimension for six breast cancer datasets. The notations are the same as in Fig. 1. (PDF 479 kb)

Additional file 2: Figure SF1. Standard estimations of intrinsic dimensionality (by Keiser rule or by broken stick distribution) of cancer datasets. (PDF 288 kb)

Additional file 3: Table ST2. Genes associated with ICA components of the METABRIC dataset, in the case when a component is driven by a small group of genes (frequently, one gene). Gene names marked in bold also drive independent components in several other breast cancer datasets and the corresponding components are reciprocally reproducible in terms of the correlation of the whole ICA-based metagenes. (XLSX 10 kb)

Additional file 4: Table ST1. Breast cancer transcriptomic datasets used for the analysis of component reproducibility in independent datasets. (XLSX 13 kb)

Additional file 5: Figure SF4. The histograms of the total number of reciprocal correlations in the correlation graph such as the one shown in Fig. 3, with and without applying the component stabilization approach. (PDF 164 kb)

Additional file 6: Figure SF5. Computational time for ICA decomposition of different orders from 2 to 100 with step 5, using compiled MATLAB fastICA implementation and stability analysis by re-computing fastICA from 100 various initial conditions. The computation is made using an ordinary laptop with Intel Core i7 processor and 16Gb of memory, in a single thread. The BRCA BEK dataset (from [27]) contains 10,000 genes in 197 samples, and the

BRCA TCGA dataset (from [28]) contains 20,503 genes in 1095 samples. The overall timing for computing all ICA decomposition with their stability analysis is 3.0 h for BRCA BEK dataset, and 6.5 h for BRCA TCGA dataset. These computations can be repeated using BIODICA software [29] (<https://github.com/LabBandSB/BIODICA>), by launching ICA computation in scanning mode. (PDF 361 kb)

Additional file 7: Figure SF3. Graph of reciprocal correlations between components computed with MSTD choice for the reduced dimension and the number of components. The size of the points reflects their stability (larger points corresponds to more stable components). The color and the width of the edges reflect the Pearson correlation coefficient. Propositions of annotations of the pseudo-cliques in the graph are made based on the comparison with previously annotated metagenes [3] and the analysis of the top contributing genes using hypergeometric test and the *toppgene* web tool [30]. (PDF 315 kb)

Abbreviations

IC: Independent Component; ICA: Independent Component Analysis

Acknowledgements

We thank Dr. Anne Biton for sharing the normalized public transcriptomics data for four breast cancer datasets. We also thank Prof. Joseph H. Lee (Columbia University) for critical reading and improving the manuscript text.

Funding

This study is supported by "Analysis of cancer transcriptome data using Independent Component Analysis" project from the budget program "Creation and development of genomic medicine in Kazakhstan" (0115RK01931) from the Ministry of Education and Science of the Republic of Kazakhstan. This work was partly supported by ITMO Cancer within the framework of the Plan Cancer 2014–2019 and convention Biologie des Systèmes N°BIO2015–01 (M5 project) and MOSAIC project.

Availability of data and materials

The results shown in this paper are in part based upon publicly available data generated by the TCGA Research Network: <http://cancergenome.nih.gov/>. The provenance of the public data used in this study is indicated in the Method section and Additional file 4: Table ST1.

Authors' contribution

UK LC EB AZ designed the study and developed the methodology, UK LC AG AM UC AZ performed the computational experiments, UK LC UC AZ wrote the manuscript, all authors read, approved and edited the manuscript.

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

Not applicable.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Author details

¹Institut Curie, PSL Research University, INSERM U900, Mines ParisTech, Paris, France. ²Laboratory of bioinformatics and computational systems biology, Center for Life Sciences, National Laboratory Astana, Nazarbayev University, Astana, Kazakhstan.

Received: 16 April 2017 Accepted: 4 September 2017

Published online: 11 September 2017

References

- Hyvärinen A, Oja E. Independent component analysis: algorithms and applications. *Neural Netw.* 2000;13(4-5):411–30.

2. Teschendorff AE, Journée M, Absil P a, Sepulchre R, Caldas C. Elucidating the altered transcriptional programs in breast cancer using independent component analysis. *PLoS Comput Biol.* 2007;3(8):e161.
3. Biton A, Bernard-Pierrot I, Lou Y, Krucker C, Chapeaublanc E, Rubio-Pérez C, et al. Independent component analysis uncovers the landscape of the bladder tumor Transcriptome and reveals insights into luminal and basal subtypes. *Cell Rep.* 2014;9(4):1235–45.
4. Gorban A, Kegl B, Wunch D, Zinovyev A. Principal Manifolds for Data Visualisation and Dimension Reduction. *Lect notes Comput Sci Eng.* 2008;58:340p.
5. Saidi SA, Holland CM, Kreil DP, MacKay DJ, Charnock-Jones DS, Print CG, et al. Independent component analysis of microarray data in the study of endometrial cancer. *Oncogene.* 2004;23(39):6677–83.
6. Zinovyev A, Kairov U, Karpenyuk T, Ramanculov E. Blind source separation methods for deconvolution of complex signals in cancer biology. *Biochem Biophys Res Commun.* 2013;430(3):1182–7.
7. Brunet JP, Tamayo P, Golub TR, Mesirov JP. Metagenes and molecular pattern discovery using matrix factorization. *Proc Natl Acad Sci U S A.* 2004;101(12):4164–9.
8. Bang-Bertelsen CH, Pedersen L, Fløyel T, Hagedorn PH, Gylvin T, Pociot F. Independent component and pathway-based analysis of miRNA-regulated gene expression in a model of type 1 diabetes. *BMC Genomics.* 2011;12:97.
9. Himberg J, Hyvärinen A, Esposito F. Validating the independent components of neuroimaging time-series via clustering and visualization. *NeuroImage.* 2004;22(3):1214–22.
10. Li Y-O, Adali T, Calhoun VD. Estimating the number of independent components for functional magnetic resonance imaging data. *Hum Brain Mapp.* 2007;28(11):1251–66.
11. Hui M, Li R, Chen K, Jin Z, Yao L, Long Z. Improved estimation of the number of independent components for functional magnetic resonance data by a whitening filter. *IEEE J Biomed Heal Informatics.* 2013;17(3):629–41.
12. Majeed W, Avison MJ. Robust data driven model order estimation for independent component analysis of fMRI data with low contrast to noise. *PLoS One.* 2014;9(4):e94943.
13. Cangelosi R, Goriely A. Component retention in principal component analysis with application to cDNA microarray data. *Biol Direct.* 2007;2:
14. Kégl B. Intrinsic dimension estimation using packing numbers. *Symp. A Q. J. Mod Foreign Lit.* 2003;15:681–8.
15. Bro R, Kjeldahl K, Smilde AK, Kiers HA. Cross-validation of component models: a critical look at current methods. *Anal Bioanal Chem.* 2008;390(5):1241–51.
16. Krumsieck J, Suhre K, Illig T, Adamski J, Theis FJ. Bayesian independent component analysis recovers pathway signatures from blood metabolomics data. *J Proteome Res.* 2012;11:4120–31.
17. Cancer Genome Atlas Research Network. The Cancer Genome Atlas Pan-Cancer analysis project. *Nat Genet.* 2013;45(10):1113–20.
18. Edgar R, Domrachev M, Lash AE. Gene Expression Omnibus: NCBI gene expression and hybridization array data repository. *Nucleic Acids Res.* 2002;30(1):207–10.
19. Giotto B, Joshi A, Freeman TC. Meta-analysis reveals conserved cell cycle transcriptional network across multiple human cell types. *BMC Genomics.* 2017;18(1):30.
20. Heng TSP, Painter MW, Consortium IGP. The immunological genome project: networks of gene expression in immune cells. *Nat Immunol.* 2008;9(10):1091–4.
21. Subramanian A, Tamayo P, Mootha VK, Mukherjee S, Ebert BL, Gillette MA, et al. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proc Natl Acad Sci U S A.* 2005;102(43):15545–50.
22. Dhivya P, Harris L. Circulating Tumor Markers for Breast Cancer Management. *Mol. Pathol. Breast Cancer.* Springer International Publishing; 2016. p. 207–18.
23. Himberg J, Hyvärinen A. ICASSO: Software for investigating the reliability of ICA estimates by clustering and visualization. *Neural Networks Signal Process. - Proc. IEEE Work.* 2003. p. 259–68.
24. R Core Team. R: A language and environment for statistical computing. R Foundation for Statistical Computing. Vienna, Austria; 2017. <https://www.R-project.org/>.
25. BIODICA docker web-page [Internet]. 2017. Available from: <https://hub.docker.com/r/auranic/biodica/>
26. Agarwal S, Lim J, Zelnik-Manor L, Perona P, Kriegman D, Belongie S. Beyond pairwise clustering. *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.* 2005. p. 838–45.
27. Bekhouche I, Finetti P, Adelaïde J, Ferrari A, Tarpin C, Charafe-Jauffret E, et al. High-resolution comparative genomic hybridization of inflammatory breast cancer and identification of candidate genes. *PLoS One.* 2011;6(2):e16950.
28. Koboldt DC, Fulton RS, McLellan MD, Schmidt H, Kalicki-Veizer J, McMichael JF, et al. Comprehensive molecular portraits of human breast tumours. *Nature.* 2012;490(7418):61–70.
29. Kairov U, Zinovyev A, Kalykhbergenov Y, Molkenov A. BIODICA GitHub page [Internet]. 2017. Available from: <https://github.com/LabBandSB/BIODICA/>.
30. Chen J, Bardes EE, Aronow BJ, Jegga AG. ToppGene suite for gene list enrichment analysis and candidate gene prioritization. *Nucleic Acids Res.* 2009;37:W305–11.

Submit your next manuscript to BioMed Central and we will help you at every step:

- We accept pre-submission inquiries
- Our selector tool helps you to find the most relevant journal
- We provide round the clock customer support
- Convenient online submission
- Thorough peer review
- Inclusion in PubMed and all major indexing services
- Maximum visibility for your research

Submit your manuscript at
www.biomedcentral.com/submit



Chapter 4

Application of Independent Component Analysis to Tumor Transcriptomes Reveals Specific and Reproducible Immune-Related Signals

Urszula Czerwinska, Laura Cantini, Ulykbek Kairov, Emmanuel Barillot, Andrei Zinovyev

Published in *Lecture Notes in Computer Science* book series (LNCS, volume 10891) at conference LVA/ICA 2018: *Latent Variable Analysis and Signal Separation*, 6 June 2018

4.1 Context

LVA/ICA conference is an interdisciplinary conference that gathers researches working on Latent Variable Analysis and Signal Separation in different fields of application. Works presented at LVA/ICA conference can be both methodological and application works. Submitted papers are reviewed by at least three members of the Technical Program Committee (TPC) or by competent additional reviewers assigned by the TPC members.

I have decided to expose my work on immune-related signals obtained using the fastICA algorithm to the signal deconvolution community in order to receive feedback from the experts of the field. It was a great chance to systematize my findings on overdecomposition of breast cancer transcriptomes and describe them in details. I did not aim to expand biological interpretation in this work given the technical character of the conference.

4.2 Description

In this work, I applied ICA to six breast cancer datasets in a way to obtain a high number of sources ($k \gg MSTD$ as described in the previous chapter). Then, I identified the sources related to the immune signals in all of the datasets. Finally, I concluded that three cell-types could be identified: T-cell, B-cell and Myeloid cells in most of the datasets.

I am using the protocol of decomposition defined in the previous chapter:

1. Compute the MSTD for each dataset
2. If number of samples is >100 then decompose to $k=100$, otherwise to the maximal possible number of components ($k = m$) (assuming that $100 \gg 25$ - the average MSTD)
3. Interpret the components with the gene enrichment methods

I also improved this over-mentioned protocol with additional steps that will be later included in [DeconICA R package](#) described in chapter 6.

1. The components of the S matrix are oriented in the direction of the *heavy tail* (the side of an ICA component with absolute higher weights) so that the *top* genes are always at the positive side.
2. The components are interpreted through correlations with two panels:
 - reference metagenes, published in [?] - the factors present in most tumor transcriptomes
 - immune cell-type signatures used by CIBERSORT [?] as pure cell-type profiles
 - reciprocity was a condition to label the components with reference metagenes and maximal correlation for immune cell-type signatures

4.3 Impact on the further work

With six independent datasets, I validated the hypothesis that using decompositions of $k \gg MSTD$. It is possible to compute signals corresponding to immune cell types in tumor transcriptomes.

I was also able to test different ways to characterize components and chose the one giving the most consistent results in many datasets. This work was an important step that enabled me to develop an R package and apply it to over 100 cancer datasets.

In the next chapter, I will show a comparison of this approach with an alternative method: Non-negative Matrix Factorisation (NMF).



Application of Independent Component Analysis to Tumor Transcriptomes Reveals Specific and Reproducible Immune-Related Signals

Urszula Czerwinska^{1,3}(✉) , Laura Cantini¹ , Ulykbek Kairov² , Emmanuel Barillot¹ , and Andrei Zinovyev¹

¹ Institut Curie, INSERM U900, PSL Research University,
Mines ParisTech, 26 rue d'Ulm, Paris, France
urszula.czerwinska@curie.fr

² Laboratory of Bioinformatics and Computational Systems Biology, Center for Life Sciences, National Laboratory Astana, Nazarbayev University, Astana, Kazakhstan

³ Center for Interdisciplinary Research, Paris Descartes University, Paris, France
<https://sysbio.curie.fr/>

AQI

Abstract. Independent Component Analysis (ICA) can be used to model gene expression data as an action of a set of statistically independent hidden factors. The ICA analysis with a downstream component analysis was successfully applied to transcriptomic data previously in order to decompose bulk transcriptomic data into interpretable hidden factors. Some of these factors reflect the presence of an immune infiltrate in the tumor environment. However, no foremost studies focused on reproducibility of the ICA-based immune-related signal in the tumor transcriptome. In this work, we use ICA to detect immune signals in six independent transcriptomic datasets. We observe several strongly reproducible immune-related signals when ICA is applied in sufficiently high-dimensional space (close to one hundred). Interestingly, we can interpret these signals as cell-type specific signals reflecting a presence of T-cells, B-cells and myeloid cells, which are of high interest in the field of oncoimmunology. Further quantification of these signals in tumoral transcriptomes has a therapeutic potential.

Keywords: Blind source separation · Unsupervised learning
Genomic data analysis · Cancer · Immunology

1 Introduction

In many fields of science (biology, technology, sociology) observations on a studied system represent complex mixtures of signals of various origins. It is known that tumors are engulfed in a complex microenvironment (TME) that critically impacts progression and response to therapy. In the light of recent findings [1],

many cancer biologists believe that the state of tumor microenvironment (in particular, the composition of immune system-related cells) defines the long-term effect of the cancer treatment.

In biological systems information is coded in a form of DNA that do not vary a lot between different individuals of the same species. In order to trigger a function in an organism, a part of the DNA is transcribed to RNA, depending on the intrinsic and extrinsic factors, and after additional modification messenger RNA (mRNA) is translated into a protein (i.e. digestive enzyme) that fulfill a role in the organism. The mRNA information (also called transcriptome) can be captured with experimental methods at high throughput (transcriptomics) and provides an approximation of the state of the studied system (i.e. a tissue).

Given the way transcriptomic data is collected, in the resulting dataset, for each observation or sample, the measured transcripts' expression (a putative gene expression that is transcribed to mRNA, and before it is translated to a protein) level is affected by a mixture of signals coming from various sources. Thus, we adopt a hypothesis that a transcriptome is a mixture of different signals (that can be biological or technical), including cell-type specific signals.

Recent works [2–4] showed that expression data from complex tissues (such as tumor microenvironment) can be used to estimate the cell-specific expression profiles of the main cellular components present in a tumor sample. This methodology is based on a linear model of a mixture of signals and their interaction and termed cell-type deconvolution. The mentioned methods take advantage of the prior knowledge (and, at the same time, heavily depend) on the specific transcriptomic signatures (characteristic genes and their weights) of cell types composing TME; therefore, they fall into supervised learning category.

A methodology using an unsupervised data decomposition was applied, so far, in the context of tumor clonality deconvolution by Roman et al. [5]. Some attempts were made to apply Non-negative Matrix factorization to transcriptomic data as well. However, they were either applied in very simplified context of *in vitro* cell mixtures [6] or without a specific focus on the immune signals [7].

In our work, we propose to apply an unsupervised method that will decompose mixture into hidden sources, which will be as independent as possible, based uniquely on data structure and without any prior knowledge. For this purpose, we apply Independent Component Analysis (ICA) [8] that solves blind source separation problem. ICA defines a new coordinate system in the multi-dimensional space such that the distributions of the data point projections on the new axes become as mutually independent as possible. To achieve this, the standard approach is maximizing the non-gaussianity of the data point projection distributions.

As a result of ICA, conventionally, data matrix X can be approximated: $X \approx AS$, where X is a matrix of data of size $m \times n$, A is a $m \times k$ matrix, $k < m$ and S is $k \times n$ matrix [9]. In our pipeline, input data matrix $n \times m$ (n genes/probes in rows and m samples in columns) is first transposed before applying ICA to $m \times n$. Thus columns of A ($m \times k$) can be named components (m -dimensional vectors) of mixing proportions for each sample m . The S matrix

($k \times n$) is transposed to $n \times k$ where rows are projections of data vectors onto the components (a k -dimensional vector for each of n data points).

ICA has been applied for the analysis of transcriptomic data for blind separation of biological, environmental and technical factors affecting gene expression [9–13].

The interpretation of the results of any matrix factorization-based method applied to transcriptomics data is done by the analysis of the resulting pairs of metagenes and metasamples, associated to each component and represented by sets of weights for all genes and all samples, respectively [7,9]. Standard statistical tests applied to these vectors can then relate a component to a reference gene set (e.g., cell cycle genes), or to clinical annotations accompanying the transcriptomic study (e.g., tumor grade). The application of ICA to multiple expression datasets has been shown to uncover insightful knowledge about cancer biology [11,14]. In [11] a large multi-cancer ICA-based metaanalysis of transcriptomic data defined a set of metagenes associated with factors that are universal for many cancer types. Metagenes associated with cell cycle, inflammation, mitochondria function, GC-content, gender, basal-like cancer types reflected the intrinsic cancer cell properties.

In our previous work, we introduced a ranking of independent components based on their stability in multiple independent components computation runs and define a distinguished number of components (Most Stable Transcriptome Dimension, MSTD) corresponding to the point of the qualitative change of the stability profile [15].

However, an interesting observation can be made employing a number of components going far beyond the MSTD ($M \gg \text{MSTD}$), that we call here *overdecomposition*. Applying this approach, one can discover more specific components that remain reproducible between independent datasets. In this work, we present results of overdecomposition with focus on the fine decomposition of the immune signal into cell-type specific signals.

In this analysis, we used a set of six independent breast cancer transcriptomic datasets (BRCATCGA [16], METABRIC [17], BRCACIT [18], BRCAEK [19], BRCAWAN [20] and BRCAEBCR [21]) to evaluate a detectability and a reproducibility of the immune cell-type related signal. Each dataset contains gene expression measured in breast tumor biopsy for a number of patients. Therefore each measured gene expression here can be a mix of expression from different cells: tumor cells, stroma cells (fibroblasts), immune cells or normal connective tissue.

Throughout this publication we employ terms: *stability*, *conservation* and *reproducibility* that we define as follows. Stability of an independent component, in terms of varying the initial starts of the ICA algorithm, is a measure of internal compactness of a cluster of matched independent components produced in multiple ICA runs for the same dataset and with the same parameter set but with random initialization. Conservation of an independent component in terms of choosing various orders of the ICA decomposition is a correlation between matched components computed in two ICA decompositions of different orders (reduced data dimensions) for the same dataset. Reproducibility of an independent

component is an (average) correlation between the components that can be matched after applying the ICA method using the same parameter set but for different datasets. We claim that if a component is reproduced between the datasets of the same cancer type, then it can be considered a reliable signal less affected by technical dataset peculiarities. If the component is reproduced in datasets from many cancer types, then it can be assumed to represent a universal cancerogenesis mechanism, such as cell cycle or infiltration by immune cells.

2 Methods

2.1 ICA Overdecomposition Procedure

Our pipeline can be described as follows. Started with six public transcriptomic data of breast cancer, we apply the fastICA algorithm [8] accompanied by the icasso package [22] to improve the components estimation and to rank the components based on their stability. In order to run the analysis we used open source BIODICA tool (ICA applied to BIOlogical Data), available from <https://github.com/LabBandSB/BIODICA>. It provides both a command line and a user-friendly Graphical User Interface (GUI) for high-performance ICA analysis, including bootstrapping and further stability analysis. It also allows the computation of MSTD index, introduced in [15]. BIODICA software links to downstream analysis enabling the interpretation of components, such as standard statistical methods, i.e. enrichment test, and non-standard methods, such as using projection on top of molecular maps (InfoSigMap, [23]). The downstream analysis was not exhaustively employed in this publication as we focused on specific immune signals.

ICA was applied to each transcriptomic dataset separately. For each analyzed transcriptomic dataset, we computed M independent components (ICs), using *pow3* nonlinearity and symmetrical approach to the decomposition. The number of dimensions was set to 100 ($M = 100$) as it is significantly greater than MSTD for these datasets (that is in the order of $M = 30$). Each component of the resulting S matrix was oriented in the direction of its heavy tail, being defined as the tail with the maximum sum of absolute weight values, so that it always has the positive sign.

2.2 Interpretation of Components

In order to confirm that we can recover expected known signals performing the overdecomposition procedure, we correlate reference metagenes with the S matrix. Correlations are performed on common genes for each component and metagene. The result was graphically represented using R package *ggplot2* [24]. An interpretation is assigned to a component only if its assignment is reciprocal. In our analysis reciprocity is defined as follows. Given correlations between the set of metagenes $M = \{M_1, \dots, M_m\}$ and S matrix $S = \{IC_1, \dots, IC_N\}$, if $S_i = argmax_k(corr(M_j, S_k))$ and $M_j = argmax_k(corr(S_i, M_k))$, then S_i

and M_j are reciprocal. In this way, the breast cancer metagenes were matched against the following set of previously defined metagenes [11] - reference metagenes: MYOFIBROBLASTS, BLCA PATHWAYS, STRESS, GC CONTENT, SMOOTH MUSCLE, MITOCHONDRIAL TRANSLATION, INTERFERON, BASALLIKE, CELL CYCLE, UROTHERIAL DIFF. Details about construction of reference metagenes and their interpretation can be found in Biton et al. 2014 [11]. The correlation plot was visualized in Cytoscape 2.8 [25].

2.3 Selecting Immune-Related Components

In order to preselect immune-related signals, we focused on all Independent Components (ICs) with Pearson correlation > 0.1 between IMMUNE metagene and ICs (columns of the S matrix). The interpretation was given using Fisher exact test on 100 top-ranked genes of each of the preselected components and Immgen [26] signatures containing in total 6467 genes of six immune cell types: $\alpha\beta$ T-cells, $\gamma\delta$ T-cells, B-cells, CD+, Myeloid cells, NK cells and four non-immune cell types: Fetal-Liver, Stem cells, Stromal cells and Pasmocytoid, 241241 signatures in total, each of 480 genes in average.

2.4 Comparing Independent Components from Different Datasets

Following the methodology developed previously in [11], the metagenes computed in two independent datasets were compared by computing a Pearson correlation coefficient between their corresponding gene weights. Since each dataset can contain a different set of genes, the correlation is computed on the genes which are common for a pair of datasets. Note that this common set of genes can be different for different pairs of datasets. The same correlation-based comparison was done with previously defined and annotated metagenes. In all correlation-based comparisons, the absolute value of the correlation coefficient was used.

3 Results

3.1 Most of Known Metagenes Can Be Found in Overdecomposed Datasets

In all six overdecomposed datasets of breast cancer, we could find major reference metagenes [11]. As an example, we present results for METABRIC dataset [17] (Fig. 1) where we can observe correlations between metagenes and all 100 ICs. For some metagenes (MYOFIBROBLASTS, INTERFERON, MITOCHONDRIAL TRANSLATION, CELL CYCLE), there is only one reciprocal and strongly (>0.3) correlated component, which can be understood as a good signal reproducibility. Some other as STRESS, BASALLIKE and SMOOTH MUSCLE can have two similarly correlated components. This is probably due to component split in higher-order decomposition. Importantly, reference metagenes were

defined in significantly lower dimensional space ($M = 25$) and as a result of high-dimensional decomposition, these signals are decomposed to more specific sources that can still be interpreted in biological terms. For few components, no strong correlations with metagenes were found (UROTHELIALDIFFERENTIATION and BLCPATHWAYS). As these metagenes are more specific to Bladder cancer, we can consider them as negative control here. Also, GC Content and IMMUNE metagenes have several corresponding components. The IMMUNE metagene is considered here as a special case as we can find several components correlated to it and, in addition, their interpretation can be interesting for biological applications. We investigate more about the immune-related components in the Subsect. 3.3.

3.2 Reproducibility of the Signals in Breast Cancer Datasets

It would be reasonable to expect that the main biological signals are characteristic for a given cancer type. Thus, they should be the same when one studies molecular profiles of different independent cohorts of patients. For this reason, we expect that for multiple datasets related to the same cancer type, the ICA decompositions should be somewhat similar; hence, reciprocally matching each other.

We correlated the ICA overdecompositions of all six datasets with each other and with the forementioned metagenes [11]. One can notice from the correlation graph (Fig. 2A), that some pseudo-cliques characterized with strong correlation coefficient (thick edges) and reciprocal (green) edges are present in the mass of low correlation coefficients edges. If the edges with correlation coefficient < 0.4 are filtered out, we can better visualize a collection of pseudo-cliques (Fig. 2B). Some of those pseudo-cliques are connected to a metagene and can be given an interpretation directly, some others would need a further investigation of the gene signature in order to attribute a meaning to them. We can see that in some pseudo-cliques not all datasets are represented. It may suggest that some signals, still reproducible, are not representative for all datasets. In order to explain, why a signal is missing, one should first interpret the signal, then try to understand the similarities or differences of samples based on provided metadata. From our previous analysis [11], the components that do not find reciprocity (absent from the pseudo-cliques) are either dataset specific or they correspond to unknown batch effects that cannot be guessed without an additional knowledge. It is remarkable that despite overdecomposition, the metagenes conceived in lower-dimensional space are highly conserved and reproducible, which suggests the overdecomposition does not diminish strong signals conceived in “optimal” dimensional space (i.e. MSTD). Of note, these datasets were produced using various technologies of transcriptomic profiling.

3.3 Three Pseudo-cliques Related to Three Immune Cell Types

To better understand the reproducibility of the immune-related signal, we extracted only components correlated with IMMUNE > 0.1 . Hence, we obtain

three strongly connected cliques (Fig. 3) and some disconnected components. We interpreted each of the ICs with an enrichment test. The results of Fisher exact test indicate mainly three cell types T-cell, B-cell and Myeloid cells with a p-value < 0.05 as indicated in the Fig. 3. While T-cell and Myeloid cell are indicated with very high certainty, the B-cell signal seems to be more complex. The results of the enrichment test for the B-cell component are less explicit as among the most enriched pathways, different cell types (T-cells and Natural Killers) are listed together with dominating B-cell signal. However, this can be explained by functional and phenotypic similarities between NK and B cells [27]. Also, T cell and B cell as they are both lymphocytes, they share common features. It is worth highlighting that definition of cell type signature is a part of ongoing debate [28] and here we use them as an indicator of possible signal definitions. Also, some ICs belonging to one pseudo-clique are correlated (with lower coefficients) with ICs from another pseudo-clique (i.e. BRCABCR IC2). It may suggest an inclination of the signal towards the other phenotype. As far as components not included in pseudo-cliques are concerned, through interpretation BRCACIT IC42 can be associated with B cells, METABRIC IC28 with Myeloid cells, BRCAWAN IC68 and BRCABEK IC27 with T-cells. Thus, the correlations of the disconnected components, even though they are low, they are most probably not spurious. Some other components not included in the pseudo-cliques like BRCAWAN IC28 and BRCABCR IC19 seem to contain stroma elements. It would be worth understanding more deeply the nature of each signal and interpret in terms of biological functions or sub-phenotypes.

4 Discussion

The overdecomposition of six breast cancer datasets, where different normalization methods and different transcriptome profiling platforms were used, showed that even in high order blind source separation, the ICA-based analysis can be reproducible between datasets. Moreover, the most stable signals are conserved and not affected by the number of dimensions. Interestingly, for some signals we can observe a split into more specific signals that can still be interpreted in biological terms. In the case of the immune-related signals, it allows robust reproduction of three main signals that form pseudo-cliques on the correlations graph in the Fig. 3. This result let us believe that ICA allows separation of signals in cancer transcriptomes in an unsupervised manner and detect the most represented immune cell-types. We found highly interesting that technically non-stable signal is found reproducible and interpretable in the six breast cancer datasets.

The question about the choice of ICA over other available blind source separation methods can be asked. We address this question more extensively in a publication in preparation comparing NMF, ICA and PCA for transcriptome BSS. From our expertise (unpublished data) NMF applied to transcriptomes can effectively separate sources and their proportions (proven in controlled mixtures of different cell types or tissues). However, when NMF was applied to noisy tumor

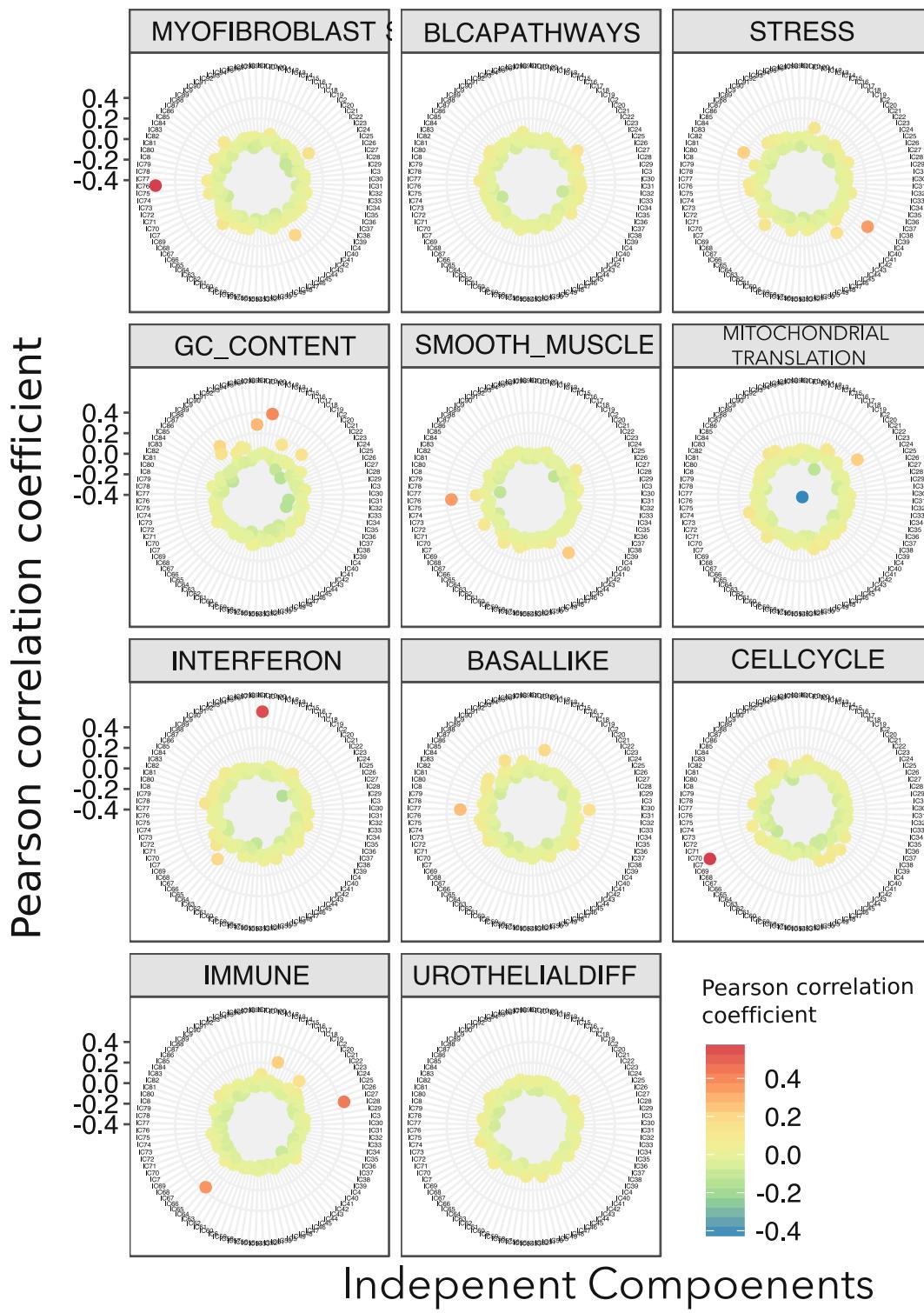


Fig. 1. Correlations between 11 metagenes [11] and 100 independent components of METABRIC dataset [17]. Each panel shows correlation coefficients between a given metagene and 100 ICs of METABRIC, the components are ordered in the same manner for all panels from 1 to 100 in a circle. For a high correlation coefficient, the point is red, for low, it is blue (see legend). (Color figure online)

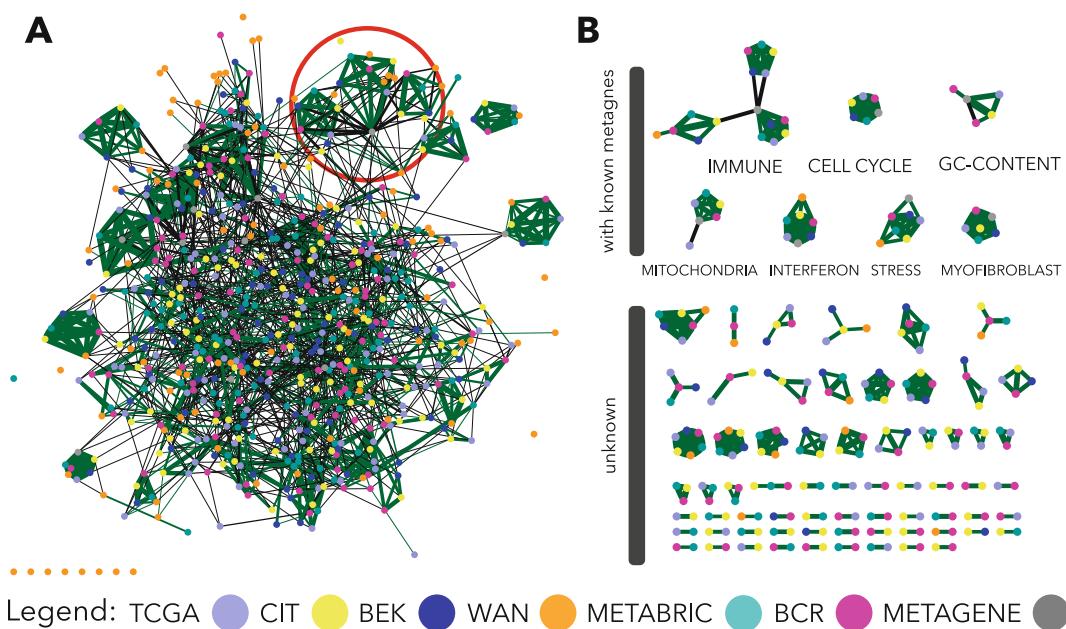


Fig. 2. Correlation plot of six tumor datasets and the reference metagenes [11] A- Correlation graph between decompositions into 100 ICs of the six transcriptomic datasets and the 11 reference metagenes. The IMMUNE metagene and related ICs in encircled; B - collection of pseudo-cliques extracted from the correlation graph A through filtering out edges of the Pearson correlation coefficient < 0.4 . They were split in two groups, the ones that are directly interpretable via their correlation with a metagene and cliques that are not related to any known metagene; The thickness of edges is proportional to the Pearson correlation coefficients, green color indicates reciprocity of edges, colors of nodes indicate dataset (see legend). (Color figure online)

transcriptomes, obtained source profiles were not highly reproducible between different datasets. Our unpublished research showed that NMF profiles are highly affected by mean gene expression. Therefore, NMF decomposition applied to breast cancer transcriptomes followed by correlation of obtained profiles did not reveal meaningful pseudo-cliques as the ICA-based analysis discussed in this article.

In order to translate our findings into real biomedical application, more time should be dedicated to analyze ICA signatures in details, to report their similarities and differences. As well as, this analysis could be applied in a pan-cancer manner to observe the reproducibility of the signal among different tumor types. Such an analysis would possibly identify components and/or genes linked with patients' survival or response to treatment and eventually, use them to compose a predictive score for tumor immune therapy outcome.

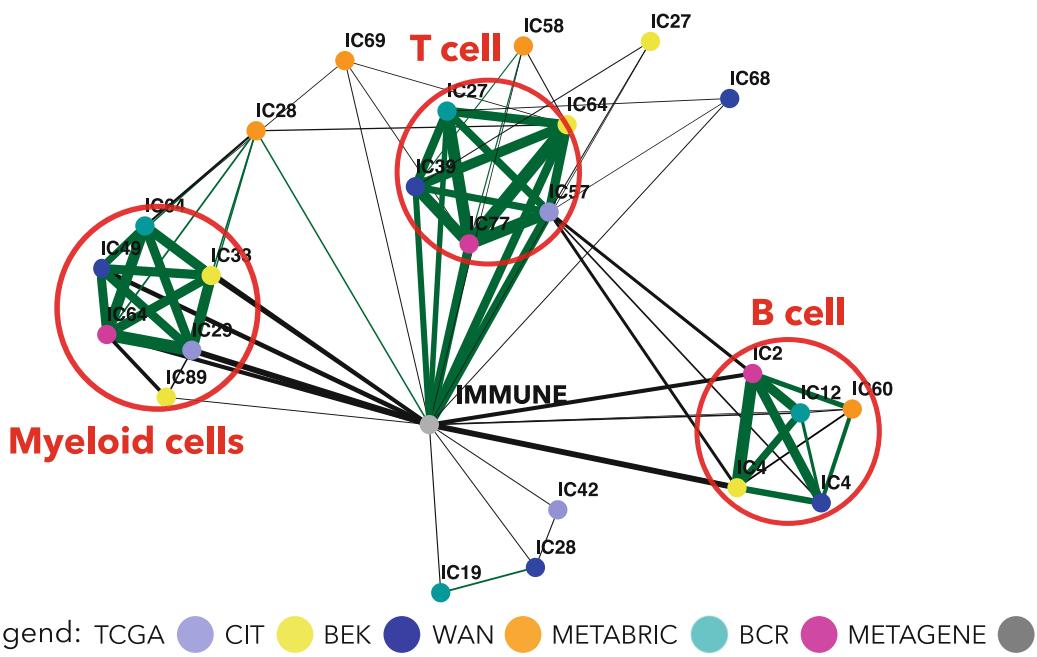


Fig. 3. Correlation graph of ICs correlated with IMMUNE metagene > 0.1 . Three pseudo-cliques are encircled and labeled according to the results of Fisher exact test. The thickness of edges is proportional to the Pearson correlation coefficients, green color indicates reciprocal edges, colors of nodes indicate dataset (see legend). (Color figure online)

5 Conclusions

We applied overcomposition into one hundred components of six transcriptomic datasets using Independent Components Analysis, a blind source separation algorithm. We used a known collection of ranked ICA-derived genetic signatures (that we call reference metagenes) to conclude that most of the signals are conserved in the higher dimensions. We noticed that some of the components split into more specific signals. Our correlation analysis of the ICA overdecompositions of the transcriptomes stated that majority of components are reproducible between datasets. Our more focused investigation of immune-related ICs demonstrated that three cell types can be named: T-cell, B-cell and myeloid cells as a reproducible source signal in the breast cancer datasets. Further interpretation of those cell-type related genomic signatures can find application in immunotherapy as predictive biomarkers for immunotherapies.

Acknowledgments. We thank Vassili Soumelis for discussions on multidimensionality of biological systems. This work has been funded by INSERM Plan Cancer N BIO2014-08 COMET grant under ITMO Cancer BioSys program and by ITMO Cancer (AVIESAN) who provided 3-year PhD grant. We would like to acknowledge as well foundation Bettencourt Schueller and Center for Interdisciplinary Research funding for the training of the PhD student.

References

1. Swartz, M.A., Iida, N., Roberts, E.W., Sangaletti, S., Wong, M.H., Yull, F.E., Coussens, L.M., DeClerck, Y.A.: Tumor microenvironment complexity: emerging roles in cancer therapy (2012)
2. Becht, E., Giraldo, N.A., Lacroix, L., Buttard, B., Elarouci, N., Petitprez, F., Selves, J., Laurent-Puig, P., Sautès-Fridman, C., Fridman, W.H., et al.: Estimating the population abundance of tissue-infiltrating immune and stromal cell populations using gene expression. *Genome Biol.* **17**(1), 218 (2016)
3. Newman, A.M., Liu, C.L., Green, M.R., Gentles, A.J., Feng, W., Xu, Y., Hoang, C.D., Diehn, M., Alizadeh, A.A.: Robust enumeration of cell subsets from tissue expression profiles. *Nat. Methods* **12**(5), 453–457 (2015)
4. Racle, J., de Jonge, K., Baumgaertner, P., Speiser, D.E., Gfeller, D.: Simultaneous enumeration of cancer and immune cell types from bulk tumor gene expression data. *eLife* **6**, e26476 (2017)
5. Roman, T., Xie, L., Schwartz, R.: Automated deconvolution of structured mixtures from heterogeneous tumor genomic data. *PLoS Comput. Biol.* **13**(10), e1005815 (2017)
6. Gaujoux, R., Seoighe, C.: Semi-supervised nonnegative matrix factorization for gene expression deconvolution: a case study. *Infect. Genet. Evol.* **12**(5), 913–921 (2012)
7. Brunet, J.P., Tamayo, P., Golub, T.R., Mesirov, J.P.: Metagenes and molecular pattern discovery using matrix factorization. *Proc. Natl. Acad. Sci.* **101**(12), 4164–4169 (2004)
8. Hyvärinen, A., Oja, E.: Independent component analysis: algorithms and applications. *Neural Netw.* **13**(45), 411–430 (2000)
9. Zinovyev, A., Kairov, U., Karpenyuk, T., Ramanculov, E.: Blind source separation methods for deconvolution of complex signals in cancer biology. *Biochem. Biophys. Res. Commun.* **430**(3), 1182–1187 (2013)
10. Teschendorff, A.E., Journée, M., Absil, P.A., Sepulchre, R., Caldas, C.: Elucidating the altered transcriptional programs in breast cancer using independent component analysis. *PLoS Comput. Biol.* **3**(8), 1539–1554 (2007)
11. Biton, A., Bernard-Pierrot, I., Lou, Y., Krucker, C., Chapeaublanc, E., Rubio-Pérez, C., López-Bigas, N., Kamoun, A., Neuzillet, Y., Gestraud, P., Grieco, L., Rebouisso, S., DeReyniès, A., Benhamou, S., Lebret, T., Southgate, J., Barillot, E., Allory, Y., Zinovyev, A., Radvanyi, F.: Independent component analysis uncovers the landscape of the bladder tumor transcriptome and reveals insights into luminal and basal subtypes. *Cell Rep.* **9**(4), 1235–1245 (2014)
12. Gorban, A., Kegl, B., Wunch, D., Zinovyev, A.: Principal Manifolds for Data Visualisation and Dimension Reduction. Lecture notes in Computational Science and Engineering, vol. 58, p. 340. Springer, Heidelberg (2008)
13. Saidi, S.A., Holland, C.M., Kreil, D.P., MacKay, D.J.C., Charnock-Jones, D.S., Print, C.G., Smith, S.K.: Independent component analysis of microarray data in the study of endometrial cancer. *Oncogene* **23**(39), 6677–6683 (2004)
14. Bang-Berthelsen, C.H., Pedersen, L., Fløyel, T., Hagedorn, P.H., Gylvin, T., Pociot, F.: Independent component and pathway-based analysis of miRNA-regulated gene expression in a model of type 1 diabetes. *BMC Genomics* **12**, 97 (2011)
15. Kairov, U., Cantini, L., Greco, A., Molkenov, A., Czerwinska, U., Barillot, E., Zinovyev, A.: Determining the optimal number of independent components for reproducible transcriptomic data analysis. *BMC Genomics* **18**(1), 712 (2017)

16. Weinstein, J.N., Collisson, E.A., Mills, G.B., Shaw, K.R.M., Ozenberger, B.A., Ellrott, K., Shmulevich, I., Sander, C., Stuart, J.M., Network, C.G.A.R., et al.: The cancer genome atlas pan-cancer analysis project. *Nat. Genet.* **45**(10), 1113 (2013)
17. Curtis, C., Shah, S.P., Chin, S.F., Turashvili, G., Rueda, O.M., Dunning, M.J., Speed, D., Lynch, A.G., Samarajiwa, S., Yuan, Y., Gräf, S., Ha, G., Haffari, G., Bashashati, A., Russell, R., McKinney, S., Aparicio, S., Brenton, J.D., Ellis, I., Huntsman, D., Pinder, S., Murphy, L., Bardwell, H., Ding, Z., Jones, L., Liu, B., Papatheodorou, I., Sammut, S.J., Wishart, G., Chia, S., Gelmon, K., Speers, C., Watson, P., Blamey, R., Green, A., MacMillan, D., Rakha, E., Gillett, C., Grigoriadis, A., De Rinaldis, E., Tutt, A., Parisien, M., Troup, S., Chan, D., Fielding, C., Maia, A.T., McGuire, S., Osborne, M., Sayalero, S.M., Spiteri, I., Hadfield, J., Bell, L., Chow, K., Gale, N., Kovalik, M., Ng, Y., Prentice, L., Tavaré, S., Markowitz, F., Langerød, A., Provenzano, E., Purushotham, A., Børresen-Dale, A.L., Caldas, C.: The genomic and transcriptomic architecture of 2,000 breast tumours reveals novel subgroups. *Nature* **486**(7403), 346–352 (2012)
18. Guedj, M., Marisa, L., De Reynies, A., Orsetti, B., Schiappa, R., Bibeau, F., MacGrogan, G., Lerebours, F., Finetti, P., Longy, M., Bertheau, P., Bertrand, F., Bonnet, F., Martin, A.L., Feugeas, J.P., Bièche, I., Lehmann-Che, J., Lidereau, R., Birnbaum, D., Bertucci, F., De Thé, H., Theillet, C.: A refined molecular taxonomy of breast cancer. *Oncogene* **31**(9), 1196–1206 (2012)
19. Bekhouche, I., Finetti, P., Adelaïde, J., Ferrari, A., Tarpin, C., Charafe-Jauffret, E., Charpin, C., Houvenaeghel, G., Jacquemier, J., Bidaut, G., Birnbaum, D., Viens, P., Chaffanet, M., Bertucci, F.: High-resolution comparative genomic hybridization of Inflammatory breast cancer and identification of candidate genes. *PLoS ONE* **6**(2), e16950 (2011)
20. Wang, Y., Klijn, J.G., Zhang, Y., Sieuwerts, A.M., Look, M.P., Yang, F., Talantov, D., Timmermans, M., Meijer-Van Gelder, M.E., Yu, J., Jatkoe, T., Berns, E.M., Atkins, D., Foekens, J.A.: Gene-expression profiles to predict distant metastasis of lymph-node-negative primary breast cancer. *Lancet* **365**(9460), 671–679 (2005)
21. Reyal, F., Rouzier, R., Depont-Hazelzet, B., Bollet, M.A., Pierga, J.Y., Alran, S., Salmon, R.J., Fourchet, V., Vincent-Salomon, A., Sastre-Garau, X., Antoine, M., Uzan, S., Sigal-Zafrani, B., de Rycke, Y.: The molecular subtype classification is a determinant of sentinel node positivity in early breast carcinoma. *PLoS ONE* **6**(5), e20297 (2011)
22. Himberg, J., Hyvärinen, A.: ICASSO: software for investigating the reliability of ICA estimates by clustering and visualization. In: Neural Networks for Signal Processing - Proceedings of the IEEE Workshop, vol. 2003, pp. 259–268, January 2003
23. Cantini, L., Calzone, L., Martignetti, L., Rydenfelt, M., Blüthgen, N., Barillot, E., Zinovyev, A.: Classification of gene signatures for their information value and functional redundancy. *npj Syst. Biol. Appl.* **4**(1), 2 (2018)
24. Wickham, H.: *ggplot2* Elegant Graphics for Data Analysis, vol. 35 (2009)
25. Shannon, P., Markiel, A., Ozier, O., Baliga, N.S., Wang, J.T., Ramage, D., Amin, N., Schwikowski, B., Ideker, T.: Cytoscape: a software Environment for integrated models of biomolecular interaction networks. *Genome Res.* **13**(11), 2498–2504 (2003)
26. Shay, T., Kang, J.: Immunological Genome Project and systems immunology (2013)

27. Kerdiles, Y.M., Almeida, F.F., Thompson, T., Chopin, M., Vienne, M., Bruhns, P., Huntington, N.D., Raulet, D.H., Nutt, S.L., Belz, G.T., Vivier, E.: Natural-Killer-like B cells display the phenotypic and functional characteristics of conventional B cells. *Immunity* **47**(2), 199–200 (2017)
28. Schelker, M., Feau, S., Du, J., Ranu, N., Klipp, E., MacBeath, G., Schoeberl, B., Raue, A.: Estimation of immune cell content in tumour tissue using single-cell RNA-seq data. *Nature Commun.* **8**(1), 2032 (2017)

Chapter 5

Comparison of reproducibility between NMF and ICA

NMF and ICA are algorithms often applied to solve blind source deconvolution problem. NMF gained popularity as a tool of transcriptomic analysis reflected in many publications [? ? ? ?]. However, none of these works compare components obtained from different datasets between each other.

The non-negativity constraint, an attractive concept in the case of non-negative transcriptome counts, may be a reason why the results of NMF decomposition are not the best candidate for our deconvolution task. I performed an analysis that demonstrates that NMF-based metagenes are less reproducible between different transcriptomic datasets than ICA-based metagenes.

5.1 Comparing metagenes obtained with NMF versus ICA

I compared the reproducibility of NMF (classical brunet version, see Section 2.3.6.2) and ICA (fastICA) through decomposition of four breast cancer datasets (BRCATCGA, METABRIC, BEK, WAN)[? ? ? , ?]]. Those datasets were selected because of their size (number of samples > 50) and because they were available in not centered format necessary for NMF.

For NMF the procedure was following:

1. data was transformed into $\log_2(x + 1)$
2. zero-rows were removed
3. the algorithm assessing cophenetic index was applied to select the optimal number of components
4. datasets were decomposed with Matlab NMF implementation from ?] into (i) number of components suggested by the cophenetic coefficient (ii) MSTD dimension (iii) 50 components (approaching overdecomposition)

5. the obtained metagenes were decorrelated from the mean using a linear regression model

For ICA, the procedure was following:

1. data were transformed into $\log_2(x + 1)$
2. transformed data were mean-centered by gene
3. our implementation of MSTD (most stable transcriptomic dimension) from [?] was used to evaluate most stable dimension
4. datasets were decomposed into (i) MSTD dimension and (ii) 50 components (approaching overdecomposition) with Matlab implementation of fastICA with icasso stabilization

I did not decompose ICA into a low number of components as we consider it as strong underdecomposition and we suspect signals would not be the most reproducible.

To define the optimal number of factors for NMF (k), I followed the strategy employed in [?] using the cophenetic coefficient which is a metric related to the stability of clusters obtained over iterative runs of NMF.

[The cophenetic coefficient] is defined as the Pearson correlation between the samples' distances induced by the consensus matrix (seen as a similarity matrix) and their cophenetic distances from a hierarchical clustering based on these very distances (by default an average linkage is used) [?]

The cophenetic distance between two observations that have been clustered is defined to be the intergroup dissimilarity at which the two observations are first combined into a single cluster. The minimum of the cophenetic coefficient values over k indicates the optimal number of factors.

Finding the best k number of factors for NMF of the biggest dataset (METABRIC) for k ranging from 2 to 50 took 30245 minutes (3 weeks). Therefore, I limited the k_{max} to 50 components (maximal number of factors) and not to 100 as initially planned.

Once, the four datasets were decomposed to MSTD, Cophenetic_{min} and 50, I proceed to the comparison of the components between datasets. I correlated all obtained metagenes with each other and with known reference metagenes [?]. We represented the results in the form of a correlation graph where nodes are metagenes from different datasets and decomposition levels, and edge width corresponds Pearson correlation coefficients (Fig 5.1).

I expected to observe a subset of components from different datasets (no matter the decomposition level) correlated with each other firmly and much less with other components in order to confirm that the signal is reproducible (can be found in several dataset) and specific (can be matched to one corresponding signal in another dataset). I used the reference components here to help with the identification of signals (labeling) of indicative nature. In ICA-based correlation of components, without applying any threshold (Fig 5.1A), some emerging clusters can be remarked

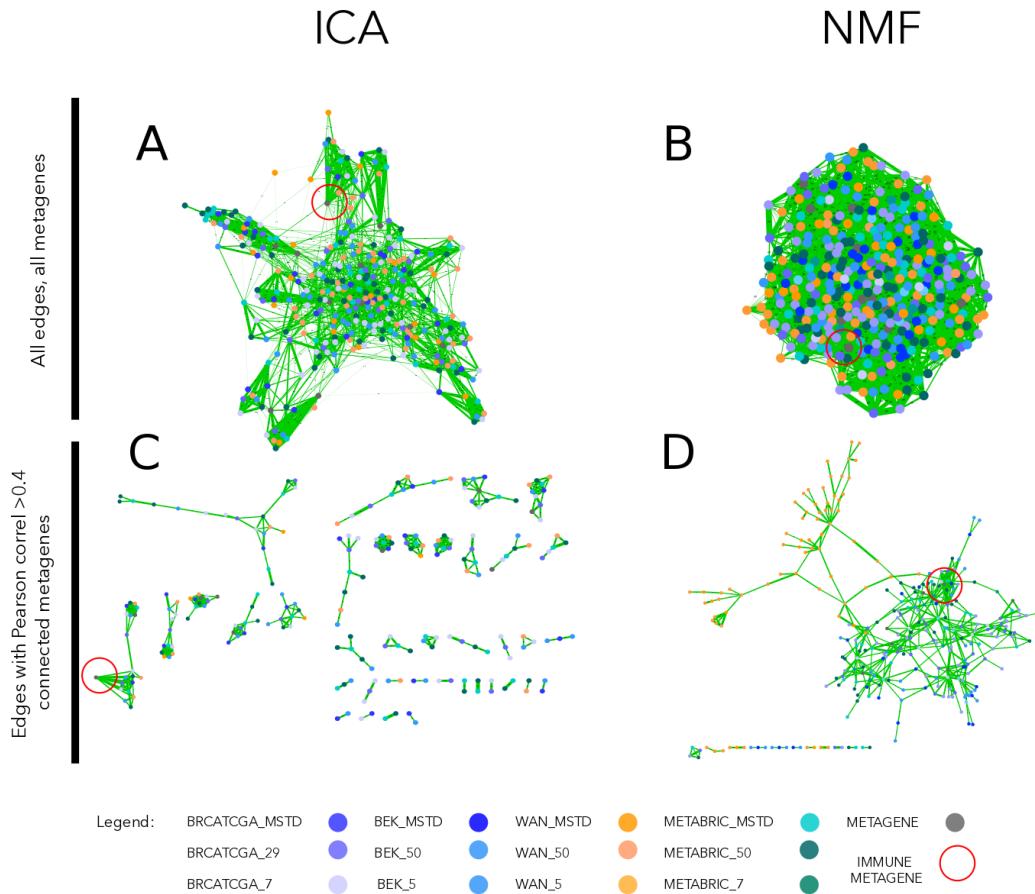


Figure 5.1: Correlation graph of ICA and NMF multiple decompositions. In the upper part of the figure (A, B) we observe the correlation graph of all metagenes (ICA or NMF-based) displayed using edge-weighted bio layout. In the lower part of the figure (C, D) we applied >0.4 thresholds to filter the edges. In the case of ICA (C), remaining nodes form pseudo-cliques, immune-related pseudo-clique is highlighted. In the case of NMF (D), components cluster by the dataset. Edges' width corresponds to Pearson correlation coefficient. Node colors correspond to the dataset from which a metagene was obtained (see legend).

and after application of >0.4 thresholds on the Pearson correlation coefficient value (Fig 5.1C) numerous pseudo-cliques emerge. While for metagenes from NMF decomposition, they are more tightly connected globally and when the threshold is applied components group by the dataset. In NMF decomposition, if it is hard to define different signals as the components seem all related to each other. We can see from (Fig 5.1D) that the IMMUNE signal is correlated >0.4 with a high number of NMF components that are also linked to some other components. In ICA (Fig 5.1C) components related to the IMMUNE metagenes form a pseudo-clique that is related only with one link to INTERFERON metagene. This makes them much more specific, and therefore the interpretation is more straightforward.

5.2 Summary

This simple analysis illustrates that NMF applied to cancer transcriptomes decomposes them to metagenes that are not selectively and specifically matching between datasets. In part, this is because NMF components are correlated with the average gene expression. Therefore, NMF can be sensitive to the normalization. However, even after the “removal” through linear regression, this phenomenon persists. It is not clear why, from a mathematical perspective, we observe such a discrepancy of interpretation of NMF and ICA components.

It is also possible that using a different method to find correct decomposition dimension (k) should be used. Ideally, different NMF implementation should be tested to verify if using different error updates can have an impact on the results.

In practice, it will not always be possible to work with the data processed in the same way. Using ICA for decomposition seems to be more straightforward, and the obtained components are easier to interpret as biological functions (thanks to the reciprocal matching) without a need to renormalize datasets.

A deepened extension of this study was performed by ?] and is available online.

Chapter 6

DeconICA: an R package for Deconvolution of omic data through Immune Components Analysis

Selected content from this chapter is a part of a publication in preparation

6.1 From blind deconvolution to cell-type quantification: general overview

In the introduction chapters 1 and 2, I have presented why there is a need to extract knowledge about the immune system from the cancer transcriptomes and how it can be done. In the result chapters 3, 4 and 5 I have presented studies of ICA application to transcriptomes, ideas for finding a way to extract the cell-specific components through *overdecomposition* procedure.

However, in the presented works, the proposed biological interpretation was not deepened. In order to enable standardized unsupervised deconvolution framework and an interpretation of obtained components as immune cell types and their quantification that is scalable, I introduce, in this chapter, a method named “DeconICA”: **Decon**volution of omic data through **I**mmune **C**omponents **A**nalysis. The method is published online on GitHub: UrszulaCzerwinska/DeconICA in the form of an R package and has a [doi number \(10.5281/zenodo.1250069\)](https://doi.org/10.5281/zenodo.1250069) for citations. In this chapter, I will describe my pipeline and the rationale behind my strategy of analysis and quantification of components. I will also briefly compare its performance in the quantifying abundance with previously published methods. The user guide for DeconICA R package is available in Annexes and online at https://urszulaczerwinska.github.io/DeconICA/DeconICA_introduction.html.

6.2 Unsupervised deconvolution

So far, I have focused on ICA-based decompositions of transcriptomes. In the chapter 5, I have compared ICA and NMF decompositions concluding that ICA-based decompositions shall be more convenient. In my work, because of my team expertise, the proven computational efficiency, interpretability, and reproducibility, I will use mostly ICA components to deconvolve tumor data (analysis described in the next chapter 6). However, the constructed interpretation pipeline can take as input any metagenes, i.e., NMF factors, convex-hull derived sources or attractor metagenes (methods described in chapter 2).

In this section, I will explain the DeconICA pipeline (Fig. 6.1) based on the stabilized fastICA overdecomposition protocol as it is resulting from our expertise and results are satisfying.

6.2.1 FastICA overdecomposition protocol

The main inputs of unsupervised deconvolution methods are data matrix X and number of output sources k . I have developed a protocol that can define k for fastICA algorithm applied to cancer transcriptome datasets and prepares the data for interpretation steps.

6.2.1.1 Data transformation

The input matrix of gene expression is transformed to $\log_2(x + 1)$ where x is a data point and then row (gene) centered: mean of each row is removed. This step is necessary for ICA algorithm.

If gene names or probes are provided, the duplicated genes are removed - the genes with higher variance are kept.

6.2.1.2 Determining k number of sources

If the number of sources in the mixture is known, k can be fixed by the user.

In case of complex mixtures of tumor transcriptomes, the input matrix of gene in rows (n) and samples (m) in columns is decomposed into $k = 100$ components if $m > 100$. If $m < 100$, then the k number is equal to the number of PCA components necessary to explain 90% of data.

If a different data type is analyzed, one can also compute MSTD (see chapter 3) of the data and redefine the overdecompositon dimension.

6.2.1.3 FastICA

The fastICA algorithm uses icasso stabilization (described in chapter 2 section 2.3.6.3) to obtain an average representation of components over the i number of iterations (by default $i = 100$) in

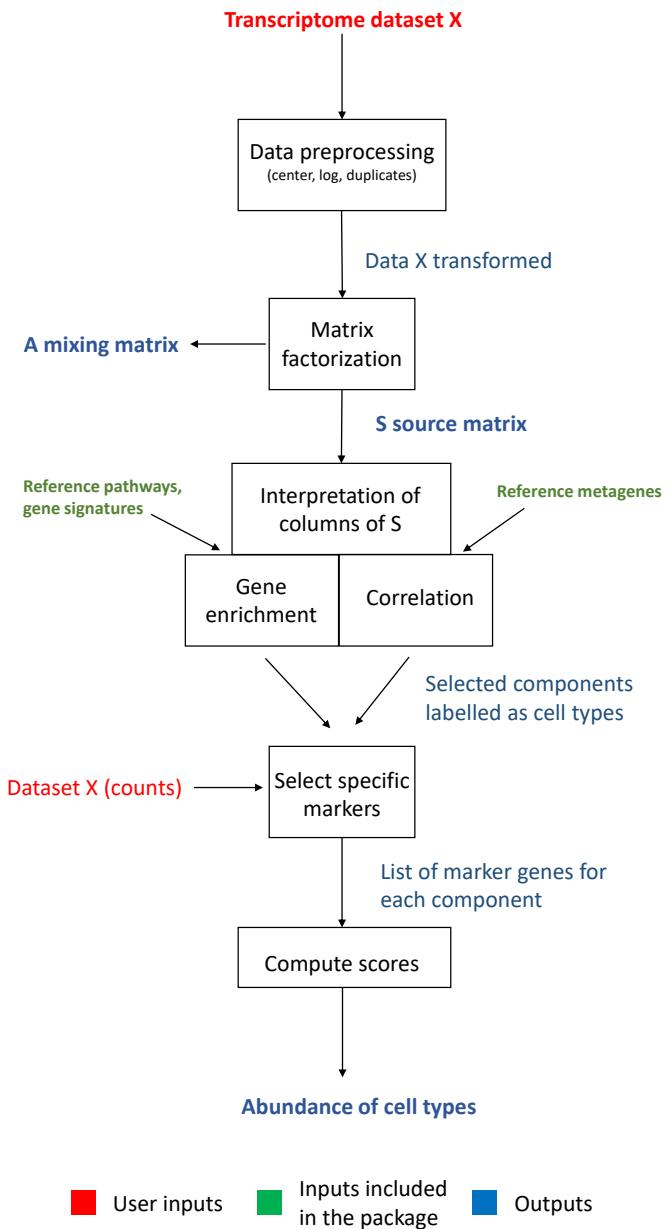


Figure 6.1: Flowchart of DeconICA method. In this flowchart steps of DeconICA are represented as boxes. Each operation corresponds to one or multiple functions in the R package. Input data X (red) is preprocessed and then decomposed into components. Interpretation of the components is performed with correlation or gene enrichment analysis (using reference materials - green). For components labeled as cell types or other important factors, abundance can be estimated using top genes of each component and computing average of counts in a non-log scale of those genes. Main outputs are (in blue), the S component matrix, labeled components, and their abundance scores.

order to buffer the effect of local minima resulting from stochastic initializations.

The stabilized version of FastICA algorithm is, so far, only available in Matlab. A custom Matlab scripts “fastica++” that I am using have initially been distributed with [BIODICA software](#) by Zinov'yev and Kairov. I have created an R interface to use it easily without any knowledge of Matlab. I also provided a detailed description on how and why use this version of fastICA as a part of [DeconICA tutorial](#) (available in Annex Z), even without Matlab software installed (through a Docker image). Although it is possible to use non-stabilized and slower R version of fastICA, it impacts the results significantly.

To remind, the input $X_{n \times m}$ matrix is therefore decomposed to $S_{k \times n}$ source matrix and $A_{m \times k}$ mixing matrix, where k is the number of components. Therefore, for example, having as an input transcriptome matrix of 20 000 genes and 150 samples, if $k = 100$ (overdecomposition), then the S will have dimensions $20\,000 \times 100$ and A matrix 100×150 .

6.2.1.4 Orienting the components

The S columns (components/sources) contain positive and negative values that the projections of data in the given dimension. The sign (positive or negative) cannot be directly interpreted. Therefore the values should be seen as an absolute value of the projection. The genes ranked top (by the absolute value), separately for positive and negative ends) are representative for the component. Usually, only one end leads to a biological interpretation. To define which end should be interpreted, I apply a simple statistical procedure (Fig. 6.2).

I plot density distribution of a component, compute a standard deviation and count how many genes are above or below a threshold of t . I define $t = 3$ standard deviation (sd). If there are no points $>t$, the threshold is lowered to 2sd , etc. The end of the independent components with a higher number of genes over the threshold is decided to be the one representing the component that we call *heavy tail*.

If the *heavy tail* is negative the component weights are multiplied by -1 in order to reverse signs.

In this way, all the representative ends of S are all positive which makes the further steps easier.

This procedure can fail if the initial sources are coming from, i.e., uniform distribution and the component S_i is symmetric. In those cases, components can be oriented for examples based on correlation with known sources (to have only positive correlations). In practical terms, there is an option to skip this step.

Generally, the orientation procedure applies to microarray and RNA-seq transcriptomes that are overdecomposed.

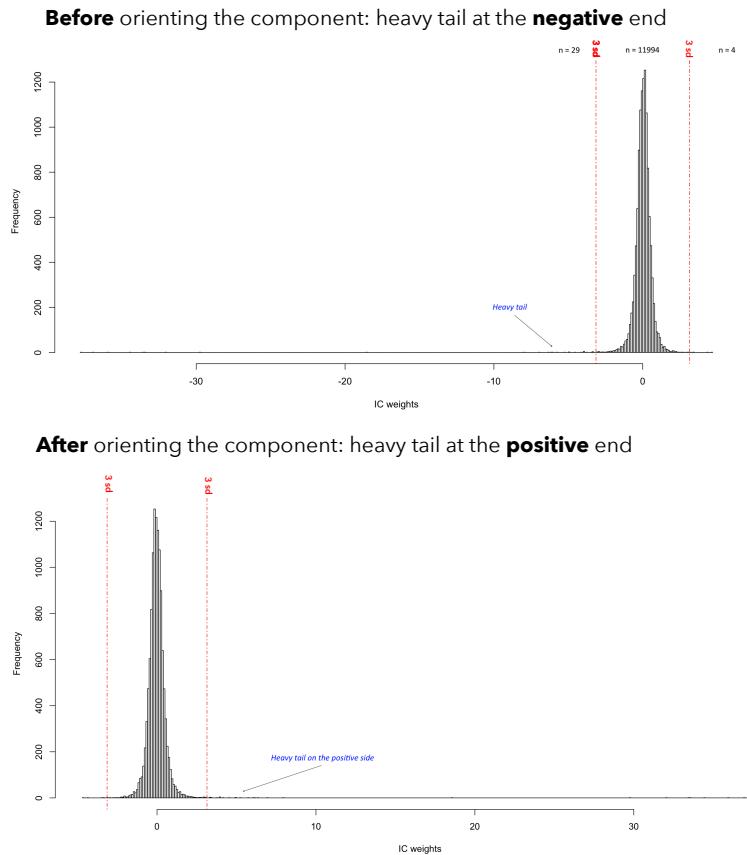


Figure 6.2: Principle of components orienting. An independent component illustrated here has more values under the threshold of - 3 standard deviations (sd) than over 3 sd (64 vs. 4). Therefore the heavy tail is on the negative side. DeconICA orients the heavy tail towards the positive side. This procedure is applied to all independent components.

6.3 Interpretation of the components

6.3.0.1 Identification of immune cell types with an enrichment test

Based on previous work of our team, a way to characterize obtained components is to select top genes and verify if they belong to a described biological pathway or a list of genes. There is a wide choice of websites (i.e., [BioGPS](#), [Toppgene](#) or [EnrichR](#)) that compares a selected list of genes with databases and computes, in different ways, an enrichment score. Often, proposed pathways or conditions are quite general, for instance, type of immune response and associated cell type would not be specified. Another way to compute enrichment is to use GSEA (described in section 2.3.3). It is possible that I will extend the package to facilitate an enrichment with GSEA (for example using the `fgsea` R package [?]). However, the essence of the interpretation though enrichment is not the score itself but the pathways/processes identified to be associated with the analyzed component.

This is why, in DeconICA package, I implemented a simple Fisher exact test (described in section 2.3.3) that computes a significance of an overlap between a list of top genes from a component and a collection of known gene sets. An ensemble of known pathways is called the universe.

I set some default parameters for the analysis: the number of top genes, minimal and maximal length of the gene list in the universe, p-value correction. I have tested the enrichment of the components in a custom collection of cell-type specific signatures published in primary research articles or as a part of deconvolution tools.

Enrichment allows to link a list of component's top genes to biological process, but the p-value depends strongly on the size of the universe and the size of gene sets. In most cases, only a small number of genes is known to be specific to a cell type. These specific genes are not always expressed in the analyzed dataset. Therefore, often false positives are found only due to technical dimension of the analysis. Depending on which compendium of signatures I used I would identify very different cell types with the enrichment test for the same component.

The enrichment methods were not conceived for small and close related gene sets. A more sophisticated solution like in xCell [?], are necessary to overcome those limitations. Thus, I would advise using basic enrichment for exploratory analysis of the general factors impacting the transcriptome rather than to identify cell types.

Looking for more robust evaluation of immune cells identification I have proposed a correlation-based interpretation.

6.3.0.2 Correlation based identification

6.3.0.3 Reciprocal match

Working with many datasets, I remarked that correlation with a reference metagene enables quickly and robustly label the components. The reference metagenes provided with DeconICA were published in the study of [?]. As these signals represent independent biological factors, one reference metagene should match one component. This is why I applied the rule of reciprocal matching to label components corresponding to reference metagenes.

Formally, the reciprocity rule is defined as follows. Given correlations between the set of reference metagenes $M = \{M_1, \dots, M_m\}$ and S source matrix $S = \{IC_1, \dots, IC_N\}$, if $S_i = argmax_k(corr(M_j, S_k))$ and $M_j = argmax_k(corr(S_i, M_k))$, then S_i and M_j are reciprocal. This rule was already applied previous publications of our group [? ? ?]. An important feature of this association is that it is not based on the strength of the correlation and hypothesise that even weak correlation can be meaningful if there is an exclusive reciprocal match between reference metagene and a component.

6.3.0.4 Maximal correlation match

As there are no established metagenes of immune cells, I used a signature matrix published in [?] LM22 containing 22 immune cells profiles reduced to 510 genes that should enable the differentiation between the cell types. I used specifically this reference as it had more genes than other similar matrices (i.e., the one from EPIC [?] contains ≈ 100 genes). The number of genes is critical because the correlation needs to be based on a minimal number of genes to be reliable. It may happen that genes present in the reference profiles are not present in studied gene matrix. Therefore, more genes in the reference set to increase the probability to interpret the data. Generally, the power of the correlation coefficient increases with the number of genes used to compute it (the overlap between the reference and the analyzed data). On the other hand, the full cell profiles (10000-20000 genes) would introduce a significant amount of noise.

To assign a reference cell type to a component, I adopted a strategy different from described in the previous section. It is known that cell type subtypes have quite similar expression profiles. Therefore, the match of one component to different subtypes of the same cell type should not be penalized. I have also remarked that sometimes closely related cell types cannot be discriminated and one component can be the most related to, i.e., T-cell and NK. Thus, the label is attributed through maximal correlation, for instance, which of all components has the highest correlation coefficient with given reference profile.

This results in the fact that each reference metagene has a component attributed. Then, the user needs to define manually if the association can be trusted from the value of the Pearson correlation coefficient. Usually, if there is one component with remarkably stronger associated to the reference cell type than others, the association can be trusted. A graphical representation (Fig. 6.3) can be help in decision making. This step should be automatized if possible in the future.

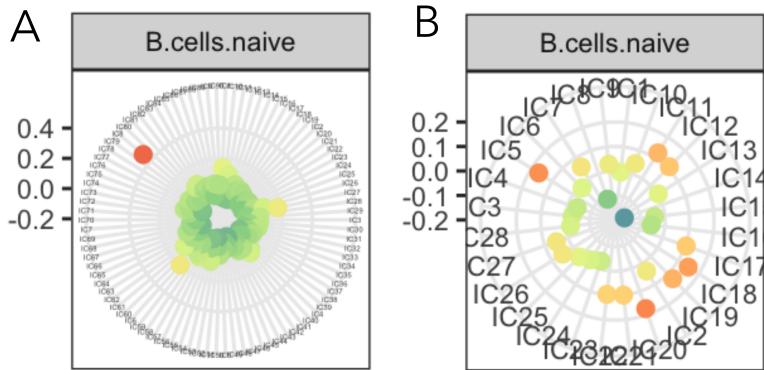


Figure 6.3: Example of successful and unsuccessful component matching to a reference. Among all components there can be one component that matches the reference profile (A) or many components that weakly matches the reference profile (B). Only in the case, 'A' the most correlated component should be labeled as B-cell.

6.4 Computing the abundance of the identified cell-types

Once the components are labeled, their contribution in each sample can be estimated. For cell type related components this contribution can be interpreted as cell-type abundance (in arbitrary units).

Before deciding on the final way in which cell-type contribution can be evaluated, I have tested different possibilities.

In theory, the A mixing matrix reflects contributions of each component. However, it reflects the contribution of both positive and negative ends of components. In my protocol, one end is selected to be representative of a biological signal. This is why A matrix scores do not reflect well abundance of the information on cell types.

One idea was to use the components as “pure cell-type expression” in a regression model (testing different regression types: SVR, quadratic programming, simple linear regression, lasso regression). I tested this approach on blood transcriptome benchmark (described in more details below). The results were acceptable, but they got outperformed by the mean of top genes approach that is included in DeconICA.

Finally, I tested the approach inspired by [?]. Selecting the most specific genes and computing arithmetic mean of the genes in the original counts matrix. This method is based on the hypothesis that those particular markers are unique to one cell type. Therefore gene expression of those genes should be proportional to the abundance of the cell type.

6.4.0.1 Defining markers

In [?] cell-type specific markers are defined based on expert knowledge and validated in gene expression data. In this work, the specific markers are generated in the unsupervised deconvolution. I adopted a hypothesis that n_{top} genes of a labeled component is a unique signature of this component. I defined the value of n_{top} empirically to be in a range of 10 to 30 genes.

6.4.0.2 Computing scores

In order to compute the scores, a mean of the selected marker genes for each component is computed. The user has a choice between arithmetical, geometric, harmonic or weighted mean. So far, the arithmetic means seem to give the best performance on the benchmark data.

6.5 Validation of the abundance estimation with DeconICA

Code used to produce the validation, and more extensive description of each step is available as a part of the [online tutorial](#) and in Annex X

6.5.1 *In silico*

First, the performance of the DeconICA was tested on simulated data. I wrote a function that produces a linear mixture of randomly generated sources of selected distribution with known proportions and possibility to set the number of marker genes. Here I generated ten sources drawn from the negative binomial distribution (10000 genes) and mix them at known proportions, adding some noise resulting in 130 mixtures (*samples*). As the ten original sources are known, a component is attributed to an original source through reciprocal correlation. I demonstrate that for each original source a matching component can be identified and using 10 top markers I estimate abundance with correlation coefficient ranging from 0.95 to 0.99 (where 1.0 is a perfect correlation), equal to average $R^2= 0.96$ (Fig 6.4).

6.5.2 *In vitro*

Previously published in [?] *in vitro* immune cell types sorted from 3 healthy donors' peripheral blood and mixed at different proportions resulting in 12 mixed samples. Following my pipeline, data is decomposed with stabilized fastICA, each of cell types finds its match and marker genes are defined as top 10 genes of each component. Then the correlation with abundance is computed. The Pearson correlation with true mixing proportions varies from 0.93 to 0.99, average $R^2= 0.94$ (Fig. 6.5).

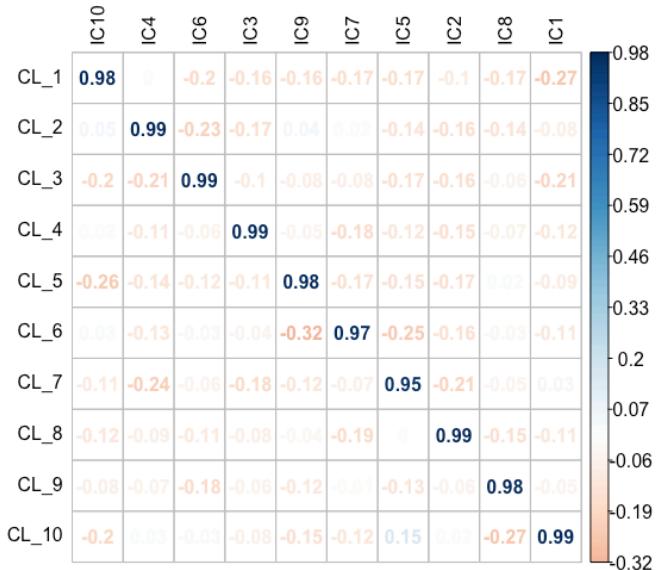


Figure 6.4: Accuracy of estimation versus true proportions in an *in silico* mixture. The simulated matrix of mixtures (10000×130) was decomposed, and obtained components were used to estimate proportions. The estimated proportions of each of 10 cell types were correlated with the known abundance values of the given cell type. The Pearson correlation coefficient values are reported in the correlation matrix.

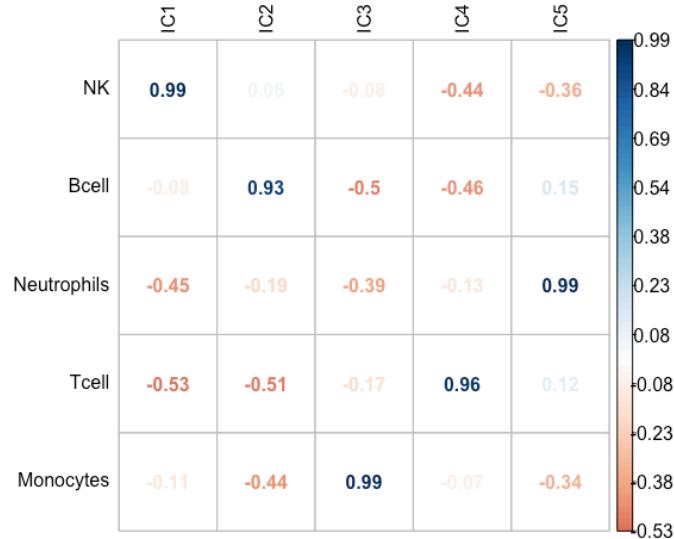


Figure 6.5: Accuracy of estimation versus true proportions in an *in vitro* mixture. Five different immune cell types from three donors were mixed at known proportions. The estimated proportions of each of 5 cell types were correlated with the known abundance values of the given cell type. The Pearson correlation coefficient values are reported in the correlation matrix.

6.5.3 PBMC transcriptome

Finally, I applied DeconICA to PBMC expression data of 104 healthy patients, paired CyTOF proportion estimation for each sample [?], processed data were shared kindly by [?]. I use MSTD to estimate the optimal dimension (39). From the correlation profiles, it can be seen that this task is remarkably more challenging than the previous tests (Fig @{fig:radarB}) and not all cell types can be perfectly matched to components. Based on maximal correlation a subset of components is labeled as immune cells and based on 10 top markers abundance is computed for B-cells, T cells CD8, T cells CD4, NK and Monocytes.

The correlation between CyTOF measured abundance and DeconICA estimated abundance varies from 0.31 to 0.77. Some cell types as T-reg, NKT, other T-cell, naive-B-cells subtypes could not be differentiated with individual components. I compared DeconICA performance with five other methods of immune cell-type deconvolution (Fig 6.7). The strategical differences between compared methods were explained in Chapter 2. All of them are recent supervised deconvolution approaches, state-of-art at the date. DeconICA performance is better or similar than the one of previously published methods for the cell types that we could identify and in average has slightly better R^2 .

It is necessary to mention that EPIC probably performs worse than expected because data were not TPM normalized. For MCPcounter CD8 T-cells were matched to “cytotoxic T-cells” and CD4 T-cells to T-cells. For CIBERSORT between naive B-cells and activated B-cells better correlation was reported as “B-cells”.

Besides, the markers discovered from data by DeconICA, are not significantly overlapping with markers used by MCPcounter or xCell (Fig. 6.8) (for EPIC and CIBERSORT list of specific cell-markers is provided as they use a basis matrix).

6.6 Summary

I developed a DeconICA method and R package that allow decomposition of omic data into components. Based on stabilized fastICA decomposition I demonstrated that DeconICA could estimate cell proportions of immune cell types in PMBC transcriptome with better performance than previously published tools without an *a priori* use of cell-type signatures. The marker genes discovered with ICA, proven to evaluate cell abundance correctly, turned out to be different from the markers used by knowledge-based supervised deconvolution methods.

Even though the first release of DeconICA is published online [?] and fully functional, some improvements can still be considered in the future. Examples using other unsupervised deconvolution methods can be included to demonstrate that the pipeline is not limited to ICA. An interactive online interface can be built with R shiny for instance. Also, more automatized label attribution and confidence of the match between a reference metagene and a component should be added. I would like also demonstrate the use of DeconICA with methylome data.

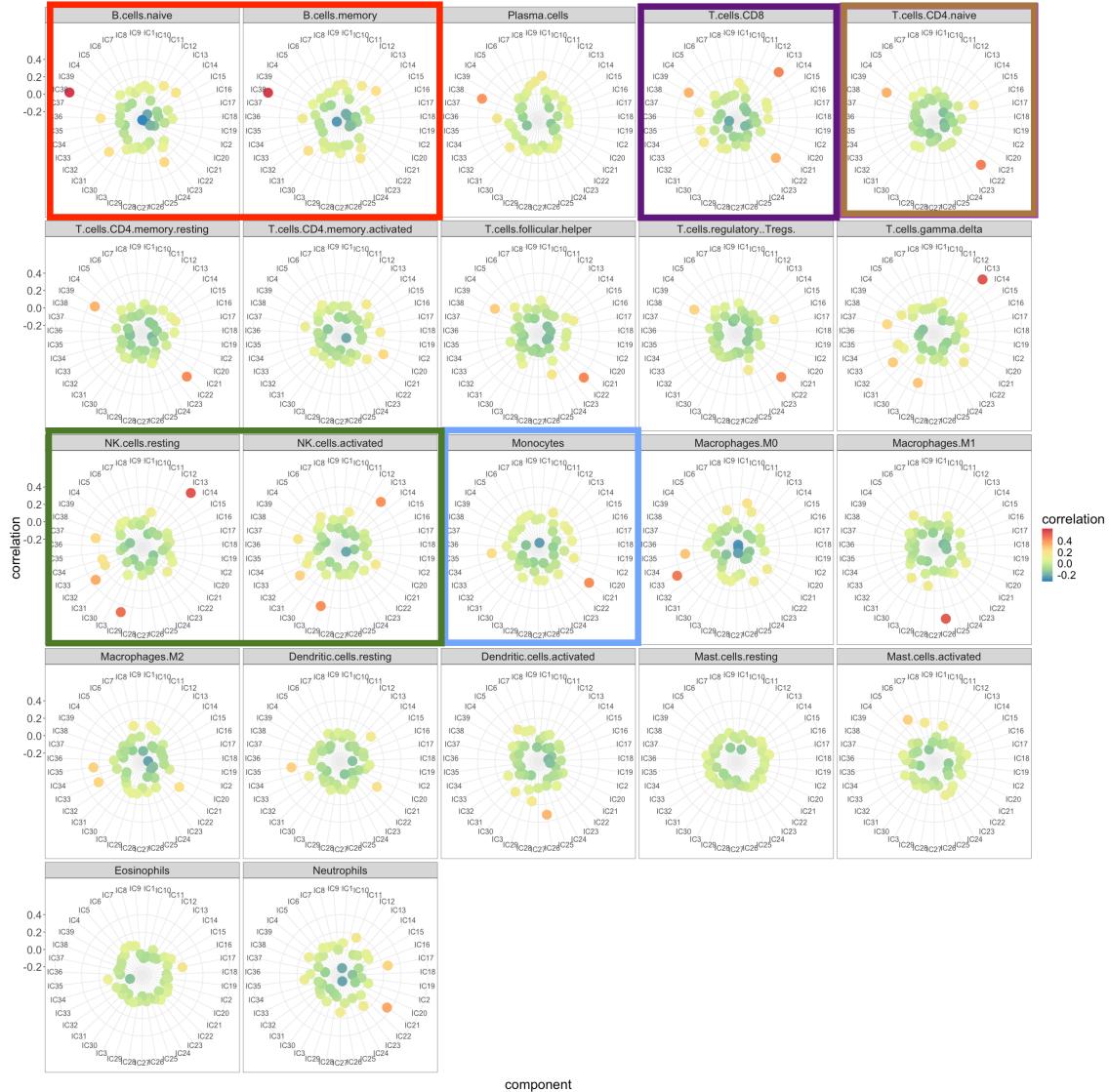


Figure 6.6: Correlation between independent components and reference immune cell-type metagenes. All correlation and reference cell types are illustrated. Surrounded by color squares are the most important panels for decision making and matching cell types measures as well with CyTOF, used for further comparison.

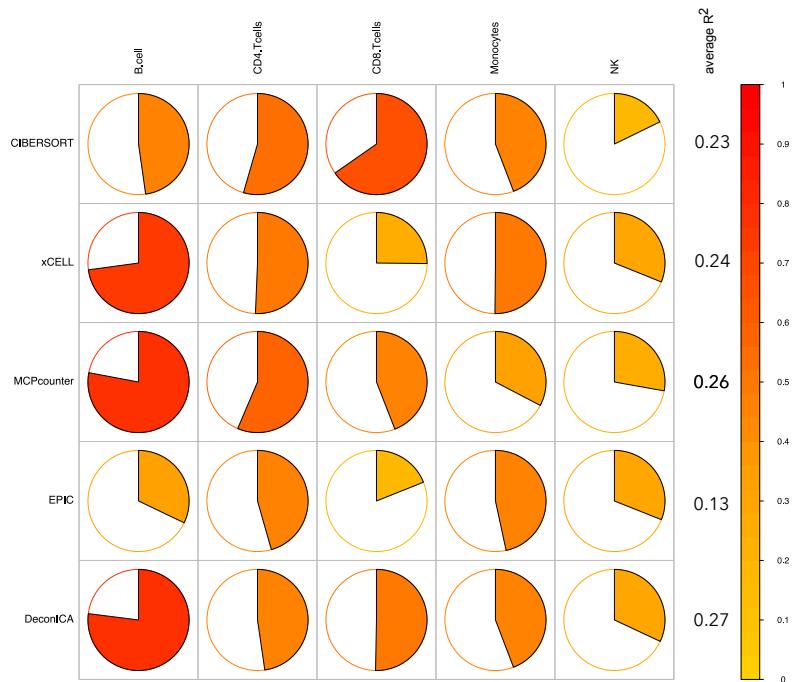


Figure 6.7: Estimation of abundance of immune cell types in PBMC transcriptome of 104 healthy donors. Five different methods (xCell [?], CIBERSORT [?], EPIC [?], MCPcounter [?], DeconICA [?]), were applied to compare estimated proportions and CyTOF measured proportions of five cell types: B-cells, CD8 T-cells, CD4 T-cells, Monocytes and NK as they were identified with DeconICA and measured proportions were available. In the case of a not exact match of cell types between the CyTOF and deconvolutions method, best correlation was reported. The average R^2 is computed as an average of R^2 for each cell types by the method. DeconICA slightly overperforms existing methods.

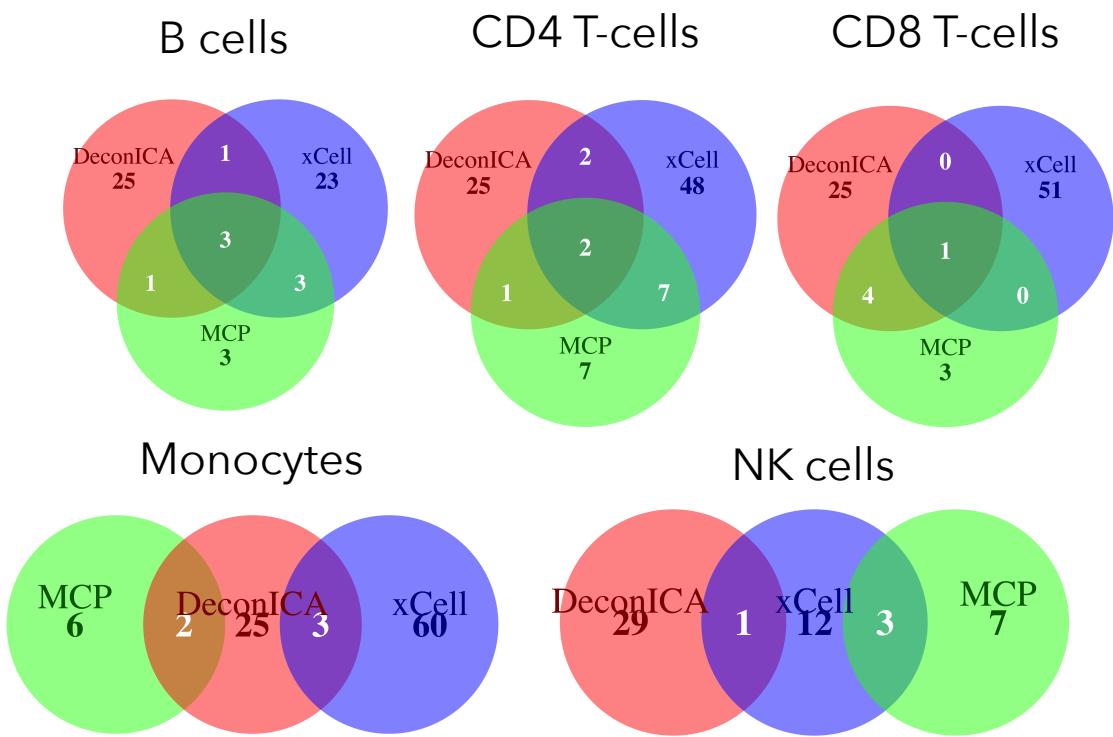


Figure 6.8: Comparison of markers used by different deconvolution methods: markers of xCell, MCPcounter, and markers discovered from data by DeconICA are compared. Venn diagrams illustrate insignificant overlap between the specific markers list for each cell type.

Main limitations of DeconICA are

- the need to work with many samples (>100)
- the interpretation that requires manual adjustments
- the number of detected cell types usually lower than other tools
- the abundance scores that cannot be directly interpreted as percentages of sample content
- it is not guaranteed to find a source of the desired cell-type

Main advantages of DeconICA are:

- the possibility to discover new markers from data
- context independence (no a priori use of blood-derived cell-type signatures)
- universality: DeconICA can identify not only cell types but also other factors governing cancer transcriptomes, can also be applied for different purposes with a little adjustment
- sequencing/ microarray platform independence
- data normalization independence
- speed (a formal benchmark is to be provided, but the analysis of the biggest available transcriptomic dataset METABRIC (1980 samples) can be analyzed within less than 1 hour)
- user-friendly form of R package, tutorials, and transparent, open-source code

So far, the tool does not have many users from outside my research group. I hope it will change once the tool is published in a scientific journal.

An ability to discover possible new marker genes of cell states can be extremely interesting in cancer context where true cell-type context-dependent signatures are not known. If this hypothesis is correct, DeconICA should not only correctly evaluate cell-type abundances but also give insight to context-dependent cell type/cell state signatures.

In the next chapter, I will apply DeconICA to >100 transcriptomic datasets of different cancer types to identify tumor-specific signatures of immune cell types, compare them and describe their features.

Chapter 7

Comparative analysis of cancer immune infiltration

Selected content of this chapter is a part of a publication in preparation

7.1 Background

7.2 Methods

7.2.1 Data sources

Bulk datasets

Single cell datasets

7.2.2 The DeconICA pipeline on bulk

Labeling components

post Data cleaning

7.2.3 The DeconICA pipeline on single cell

7.3 Results

Global image with

7.4 Discussion

7.5 Conclusions

7.6 Supplementary

7.6.1 Tables

Table 7.1: Some capiton.

ID	Name	Cancer.type	Samples	Genes	Components	Source	Normalization	Technology	PMID
1	ACC	adrenoc	78	20501	54	TCGA	custom	several	NA
2	adrenocortical-GSE10927	adrenoc	65	20621	36	GSE10927	quantile-normalized and log transformed as described	Affymetrix HG_U133_plus_2 arrays	19147773
3	AML-GSE6891	AML	536	32194	100	GSE6891	MAS5.0	Affymetrix HG-U133 plus 2	20522712
4	Bild	lung	111	33193	90	GSE3141	custom (see Bild et al.)	Affymetrix Human U133 2.0 plus arrays	16273092
5	bladder-CIT	bladder	85	32194	56	E-MTAB-1940	custom	Affymetrix GeneChip Human Genome U133 Plus 2.0	24142880
6	bladder-Kim	bladder	61	24533	28	GSE13507	quantile normalization, log2-transformed and median-centered across samples	Illumina human-6 v2.0 expression beadchip	20059769
7	bladder-Riester	bladder	78	32194	49	GSE31684	GCRMA	Affymetrix Human Genome U133 Plus 2.0 Array	22228636
8	bladder-Sjodahl	bladder	93	18581	62	GSE32894	median scaling	Illumina HumanHT-12 V3.0 expression beadchip	22553347
9	BLCA	bladder	408	20501	100	TCGA	custom	Illumina HiSeq	24476821

Table 7.1: Some capitolon. (continued)

ID	Name	Cancer.type	Samples	Genes	Components	Source	Normalization	Technology	PMID
10	Brambilla	lung	334	32396	100	?	?	hgU133plus2	?
11	BRCA	breast	1085	20501	100	TCGA	custom	Affymetrix HG-U133A	23000897
12	BRCABCR	breast	1127	6837	100	Fusion of 6 datasets from GEO: GSE6532, GSE3494, GSE1456, GSE7390, GSE5327, and ArrayExpress E-TABM-158	see methods of Reyal et al., 2011	Affymetrix HG-U133A	21655258
13	BRCABEK	breast	197	21755	100	GSE23720	GCRMA	Affymetrix HG-U133Plus2.0	21339811
14	breast-Bos	breast	204	32194	100	GSE12276	GeneSpring 7.2	Affymetrix HG-U133A	19421193
15	breast-CIT	breast	537	32194	100	ArrayExpress E-MTAB-365	GCRMA	Affymetrix HG-U133Plus2.0	21785460
16	breast-Loi	breast	327	19789	100	gse6532	custom	Affymetrix Human Genome U133A	NA
17	breast-Pawitan	breast	159	19789	100	GSE1456	global mean method	Affymetrix Human Genome U133A and U133B Array	16280042
18	Broet	lung	72	33193	46	GSE10445	custom	Affymetrix Human Genome U133 Plus 2.0 Array	20810387
19	CESC	cervical ade-nocarcinoma	303	20501	100	TCGA	custom	Illumina HiSeq 2000	28112728
20	Chang	lung	90	14376	52	GSE14814	RMA	Affymetrix Human Genome U133A Array	20823422
21	CHOL	liver	36	20501	25	TCGA	custom	Illumina HiSeq 2000 Genome Analyzers	28297679
22	COAD	colorect	278	20501	100	TCGA	custom	Affymetrix Human Genome U133 Plus 2.0 Array	22810696
23	colorectal-CIT	colorect	566	32194	100	GSE39582	ComBat	Affymetrix Human Genome U133 Plus 2.0 Array	23700391
24	colorectal-Jorissen	colorect	290	32194	100	GSE14333	custom	Affymetrix Human Genome U133 Plus 2.0 Array	19996206
25	colorectal-Smith	colorect	101	32194	61	GSE17538	NA	NA	NA
26	colorectal-Sousa	colorect	90	32194	58	GSE33113	fRMA	Affymetrix Human Genome U133 Plus 2.0 Array	22496204

Table 7.1: Some capiton. (continued)

ID	Name	Cancer.type	Samples	Genes	Components	Source	Normalization	Technology	PMID
27	Ding	lung	75	33193	51	GSE12667	custom	Affymetrix Human Genome U133 Plus 2.0 Array	18948947
28	DLBC	DLBCL	48	20501	34	TCGA	custom	Illumina HiSeq 2000 Genome Analyzers	NA
29	DLBCL-GSE10846	DLBCL	414	32194	100	GSE10846	NA	Affymetrix Human Genome U133 Plus 2.0 Array	21546504
30	DLBCL-GSE12195	DLBCL	83	32194	50	GSE12195	GenePattern	Affymetrix Human Genome U133 Plus 2.0 Array	28314854
31	DLBCL-GSE57611	DLBCL	148	13239	100	GSE57611	custom	Affymetrix Human Genome U133A Array	25042405
32	ESCA	esophageal carcinoma	182	20501	100	TCGA	custom	Illumina HiSeq 2000 Genome Analyzers	28052061
33	ewing-GSE34620	ewing	117	32194	75	GSE34620	gcrma	Affymetrix Human Genome U133 Plus 2.0 Array	22327514
34	gastric-GSE15081	gastric	141	13105	100	GSE15081	custom	Hitachisoft AceGene Human Oligo Chip 30K1 Chip Version	20012501
35	gastric-GSE29272	gastric	134	13239	100	GSE29272	custom	Affymetrix Human Genome U133A Array	24867265
36	gastric-GSE35809	gastric	70	32194	43	GSE35809	custom	Affymetrix Human Genome U133 Plus 2.0 Array	23684942
37	gastric-GSE57303	gastric	70	32194	50	GSE57303	custom	Affymetrix Human Genome U133 Plus 2.0 Array	24935174
38	GBM	glioma	760	20501	100	TCGA	custom	several	24120142
39	glioma-GSE16011	glioma	284	32194	100	GSE16011	custom	Affymetrix GeneChip Human Genome U133 Plus 2.0 Array	19920198
40	glioma-GSE4290	glioma	180	32194	100	GSE4290	custom	Affymetrix Human Genome U133 Plus 2.0 Array	16616334
41	glioma-GSE4412	glioma	85	19789	44	GSE4412	dCHIP	Affymetrix Human Genome U133A and U133B Array	15374961

Table 7.1: Some capitol. (continued)

ID	Name	Cancer.type	Samples	Genes	Components	Source	Normalization	Technology	PMID
42	glioma-GSE7696	glioma	80	32194	50	GSE7696	genewise-mean-centered, the log-scale robust multi-array average normalized	Affymetrix Human Genome U133 Plus 2.0 Array	21642372
43	glioma-Rembrandt	glioma	534	32194	100	https://caintergatc	MAS5	Affymetrix HG U133 v2.0 Plus	19208739
44	GSE13067	colorect	74	20307	47	GSE13067	MAS5.0	Affymetrix Human Genome U133 Plus 2.0 Array	19088021
45	GSE13294	colorect	155	20307	100	GSE13294	MAS5.0	Affymetrix Human Genome U133 Plus 2.0 Array	19088021
46	GSE20916	colorect	100	20307	47	GSE20916	GCRMA	Affymetrix Human Genome U133 Plus 2.0 Array	20957034
47	GSE21050	leiomyo	85	32194	56	GSE21050	GCRMA	Affymetrix Human Genome U133 Plus 2.0 Array	20581836
48	GSE21050	liposarco	62	32194	43	GSE21050	GCRMA	Affymetrix Human Genome U133 Plus 2.0 Array	20581836
49	GSE21050	undiff-pleom-sarcom	136	32194	100	GSE21050	GCRMA	Affymetrix Human Genome U133 Plus 2.0 Array	20581836
50	GSE2109	colorect	277	20307	100	GSE2109	custom	Affymetrix Human Genome U133 Plus 2.0 Array	NA
51	GSE21122	liposarco	89	13239	53	GSE21122	RMA	Affymetrix Human Genome U133A Array	20601955
52	GSE30929	liposarco	140	13239	100	GSE30929	RMA	Affymetrix Human Genome U133A Array	21335544
53	GSE33382	osteosar	84	24934	51	GSE33382	custom	Illumina human-6 v2.0 expression beadchip	23688189
54	GSE35896	colorect	62	19040	40	GSE35896	RMA	Affymetrix Human Genome U133 Plus 2.0 Array	23272949
55	GSE37892	colorect	130	20307	100	GSE37892	GCRMA	Affymetrix Human Genome U133 Plus 2.0 Array	22917480

Table 7.1: Some capiton. (continued)

ID	Name	Cancer type	Samples	Genes	Components	Source	Normalization	Technology	PMID
56	hepato-ETABM36	hepato	57	13239	38	E-TABM-36	RMA	Affymetrix GeneChip Human Genome HG-U133A	17187432
57	hepato-GSE62232	hepato	82	32194	54	GSE62232	custom	Affymetrix Human Genome U133 Plus 2.0 Array	25822088
58	hepato-GSE9843	hepato	91	32194	55	GSE9843	custom	Affymetrix Human Genome U133 Plus 2.0 Array	21324318
59	HNSC	HNSCC	515	20501	100	TCGA	custom	several	25631445
60	HNSCC-CIT	HNSCC	98	32194	68	E-TABM-302	RMA	Affymetrix GeneChip Human Genome U133 Plus 2.0	18679425
61	HNSCC-GSE39366	HNSCC	138	17093	100	GSE39366	loess normalization	Agilent-UNC-custom-4X44K	23451093
62	Jacob	lung	461	14376	100	GSE68465	MAS 5.0	Affymetrix Human Genome U133A Array	18641660
63	KICH	kidney	65	20501	42	TCGA	custom	Illumina HiSeq 2000 Genome Analyzers	29617669
64	kidney-CIT	kidney	57	20452	35	E-MTAB-3267	custom	Affymetrix GeneChip Human Gene 1.0 ST Array	25583177
65	kidney-GSE14994	kidney	59	13239	36	GSE14994	RMA	Affymetrix HT Human Genome U133A Array	19470766
66	kidney-GSE3538	kidney	177	2275	100	GSE3538	custom	Agilent	16318415
67	KIRC	kidney	515	20501	100	TCGA	custom	several	29617669
68	KIRP	kidney	285	20501	100	TCGA	custom	several	29617669
69	Kuner	lung	58	33193	41	GSE10245	custom	Affymetrix Human Genome U133 Plus 2.0 Array	18486272
70	Landi	lung	107	14376	53	GSE10245	custom	Affymetrix Human Genome U133A Array	18297132
71	Lee	lung	138	33193	100	GSE8894	GCRMA	Affymetrix Human Genome U133 Plus 2.0 Array	19010856
72	LGG	glioma	514	20501	100	TCGA	custom	several	26061751
73	LIHC	hepato	368	20501	100	TCGA	custom	Illumina HiSeq 2000 Genome Analyzers	28622513
74	LUAD	lung	594	20501	100	TCGA	custom	several	25079552

Table 7.1: Some capitolon. (continued)

ID	Name	Cancer.type	Samples	Genes	Components	Source	Normalization	Technology	PMID
75	lung-ADK-Takeuchi	lung	90	16339	55	GSE11969	custom	Agilent Homo sapiens 21.6K custom array	16549822
76	lung-ADK-Tomida	lung	117	30387	79	GSE13213	custom	Agilent-014850 Whole Human Genome Microarray 4x44K G4112F	19414676
77	lung-SQC-Wilkerson	lung	56	17028	36	GSE17710	normexp background correction and loess normalization	Agilent-UNC-custom-4X44K	20643781
78	LUSC	lung	488	20501	100	TCGA	custom	several	22960745
79	medulloblastoma-GSE10327	medulloc	62	32194	37	GSE10327	GCRMA	Affymetrix Human Genome U133 Plus 2.0 Array	18769486
80	medulloblastoma-GSE28245	medulloc	64	10814	37	GSE28245	custom	Agilent-014850 Whole Human Genome Microarray 4x44K G4112F	21911727
81	medulloblastoma-GSE37418	medulloc	76	32194	46	GSE37418	custom	Affymetrix Human Genome U133 Plus 2.0 Array	22722829
82	medulloblastoma-GSE49243	medulloc	73	32194	46	GSE49243	MAS 5.0	Affymetrix Human Genome U133 Plus 2.0 Array	24871706
83	melanoma-ETABM1	melanor	83	14764	72	E-TABM-1	custom	Agilent Whole Human Genome Oligo Microarray 012391 G4112A	16595783
84	melanoma-GSE53118	melanor	79	17494	44	GSE53118	lumi	Illumina HumanWG-6 v3.0 expression beadchip	22931913
85	MESO	mesothi	87	20501	58	TCGA	custom	Illumina HiSeq 2000 Genome Analyzers	?
86	mesothelioma-GSE29354	mesothi	53	13238	34	GSE29354	custom	Affymetrix Human Genome U133A Array	21642991
87	METABRIC	breast	1980	24360	100	EGAS00000000008	custom	Illumina HT-12 v3	22522925
88	multipleMyeloma-GSE24080	multiple	559	32194	100	GSE24080	custom	Affymetrix Human Genome U133 Plus 2.0 Array	20676074
89	multipleMyeloma-GSE9782	multiple	264	13239	100	GSE9782	MASS5.0	Affymetrix Human Genome U133A and U133B Array	17185464

Table 7.1: Some capiton. (continued)

ID	Name	Cancer.type	Samples	Genes	Components	Source	Normalization	Technology	PMID
90	neuroblastoma-GSE49710	neurobl	498	19698	100	GSE49710	custom	Agilent-020382 Human Custom Microarray 44k	25633159
91	OV	ovarian	591	20501	100	TCGA	custom	several	21720365
92	ovarian-GSE13876	ovarian	157	15942	100	GSE13876	Quantile normalization was applied to log2-transformation	Operon human v3 ~35K 70-mer two-color oligonucleotide microarrays.	19192944
93	ovarian-GSE49997	ovarian	194	16725	100	GSE49997	custom	ABI Human Genome Survey Microarray Version 2	22497737
94	ovarian-GSE6008	ovarian	99	13239	66	GSE6008	custom	Affymetrix Human Genome U133A Array	27538791
95	ovarian-GSE9891	ovarian	243	32194	100	GSE9891	custom	Affymetrix Human Genome U133 Plus 2.0 Array	18698038
96	PAAD	pancrea	177	20501	100	TCGA	custom	Illumina HiSeq 2000 Genome Analyzers	28810144
97	pancreas-GSE21501	pancrea	132	19724	100	GSE21501	Lowess normalization	Agilent-014850 Whole Human Genome Microarray 4x4K G412F	20644708
98	pancreas-GSE36924	pancrea	91	30838	56	GSE36924	CPM and log2 transformed	Illumina HumanHT-12 V4.0 expression beadchip	26909576
99	PCPG	pheochri	178	20501	100	TCGA	custom	Illumina HiSeq 2000 Genome Analyzers	28162975
100	Potti	lung	198	14326	100	GSE3593	RMA	Affymetrix Human Genome U133A Array	21366430
101	PRAD	prostate	494	20501	100	TCGA	custom	Illumina HiSeq 2000 Genome Analyzers	26544944
102	prostate-GSE21034	prostate	131	17008	100	GSE21034	custom	Affymetrix Human Exon 1.0 ST Array	20579941
103	prostate-GSE25136	prostate	79	13239	48	GSE25136	custom	Affymetrix Human Genome U133A Array	19343730
104	prostate-GSE6956	prostate	69	13239	38	GSE25136	raw	Affymetrix Human Genome U133A Array	19343730
105	Raponi	lung	130	14376	100	GSE4573	MAS 5	Affymetrix Human Genome U133A Array	16885343
106	READ	rectum	90	20501	60	TCGA	custom	several	22810696

Table 7.1: Some capitolon. (continued)

ID	Name	Cancer.type	Samples	Genes	Components	Source	Normalization	Technology	PMID
107	SARC	sarcoma	255	20501	100	TCGA	custom	Illumina HiSeq 2000 Genome Analyzers	29100075
108	SKCM	melanoma	103	20501	66	TCGA	custom	Illumina HiSeq 2000 Genome Analyzers	26091043
109	STAD	gastric	401	20501	100	TCGA	custom	Illumina HiSeq 2000 Genome Analyzers	25079317
110	Su	lung	66	14376	28	NA	custom	Affymetrix Human Genome U133A Array	7540040
111	TGCT	testicular	149	20501	100	TCGA	custom	Illumina HiSeq 2000 Genome Analyzers	NA
112	THCA	thyroid	501	20501	100	TCGA	custom	Illumina HiSeq 2000 Genome Analyzers	25417114
113	THYM	THYM	120	20501	100	TCGA	custom	Illumina HiSeq 2000 Genome Analyzers	29438696
114	Tzu-lu	lung	120	33193	100	GSE19804	Partek	Affymetrix Human Genome U133 Plus 2.0 Array	20802022
115	UCEC	uterine	173	20501	100	TCGA	custom	several	23636398
116	UCS	uterine	57	20501	42	TCGA	custom	several	23636398
117	UVM	melanoma	80	20501	47	TCGA	custom	Illumina HiSeq 2000 Genome Analyzers	28810145
118	WAN	breast	286	12993	100	GSE2034	GCRMA	Affymetrix HG-U133a	15721472
119	Zhou	lung	79	27634	54	GSE4824	NA	NA	16843264

Table 7.2: asez

dataset	type	normal.cells	cancer.cells	total.cells	B.cells	T.cells	macrophages	endothelial.cell	mast.cell	DC	fibroblasts	NK	patients	PMID
Tirosh et al.	Melanoma	3256	1257	4645	515	2068	126	65	0	0	61	52	19	27124452
Li et al.	CRC + normal mucosa	215	375	215	35	45	29	6	4	0	26	0	11	28319088
Chung et al.	Breast+ lymph nodes	175	317	515	83	54	38	0	0	0	23	NA	11	28474673
Zheng et al.	Liver + PBMC+ Adjacent	1939	3124	5063	0	5063	0	0	0	0	0	0	6	28622514
Sidharth et al.	Head and Neck	3363	2215	5902	138	1237	98	0	120	51	1440	0	18	29198524

Chapter 8

A multiscale signalling network map of innate immune response in cancer reveals signatures of cell heterogeneity and functional polarization

Maria Kondratova^{*}, Urszula Czerwinska^{*}, Nicolas Sompairac, Sebastian D Amigorena, Vassili Soumelis, Emmanuel Barillot, Andrei Zinovyev and Inna Kuperstein

^{*} contributed equally

Under review in *Nature Communications*

8.1 Context

The intra- and intercellular signaling pathways are a broad subject of biological research. It is known that in cancer disease, important signaling pathways get altered. These phenomena got described in the field-breaking [?] publication Hallmarks of cancer. Many pathways altered in cancer determine how cell get out of the ‘healthy state’ and become invasive, immortal and deleterious. Researchers performing metabolic, proteomic and genetic experiments can measure the interactions between different molecules and link with observed phenotype. This knowledge is collected in databases, of so-called, protein-protein interactions or pathway databases (i.e. HPRD [?], STRING [?], REACTOME [?]), some including metabolic interactions as well RECON [?], KEGG[?]. These databases can, besides the experimental knowledge, contain computationally inferred interactions based on text mining or data inference. It is a usual practice as well to include interactions observed in different animal species (Human, mice, yeast), or in different states (healthy, cancer, infection). Therefore, it is not trivial to retrieve relevant information for a given

organism in a given state. In our group, there was created an Atlas of Cancer Signaling Networks (ACSN) that contains manually curated interactions retrieved from cancer-specific literature, to create a compendium of knowledge that is specific to tumor cells. The created database has a number of additional features facilitating the content exploration (google map based semantic zooming), hierarchical content organization (pathways, modules, maps), manually designed layout facilitating interpretation and allowing projection of proteomic and genomic data into the existing pathways to create, so-called, molecular portraits illustrating state of the represented pathways (activation/inhibition) in the data.

However, given the importance of the TME in the cancer progression and response to treatment, the intracellular cancer pathways are not enough to have a system-level view of the cancer data. The evaluation of the polarization status within the subtle innate immune cell subpopulations in TME is essential for the improvement of immunotherapy. This is why, in my team, we developed TME-related maps that will be soon a part of ACSN 2.0. The first part of TME maps collection was related to the innate immunity including NK, Macrophages and DC maps which form together, with additional intercellular interactions, an innate immunity meta map. With respect to existing databases of immune signaling networks, the innate immune maps we propose are cancer-focused and minutely manually curated, which make our resource the first of its kind.

In this publication which I am a co-author, in the first place, we describe the details and strategy of the innate immunity maps creation. In the second place, we demonstrate with scRNA-seq transcriptomic profiles of NK and Macrophages in Metastatic Melanoma how this maps can be used to understand immune cells heterogeneity better.

I participated mostly in the second part of the work and as well as in figures design and article writing.

8.2 Description

The created innate immune meta map contains 1466 nodes among which there are 582 proteins, 1084 biochemical reactions based on 837 cell type-specific and cancer-related articles. The map has a multidimensional hierarchical structure (Fig.3 ?])

- right-left axis: anti- and pro-tumor polarisation
- up-down axis: signaling pathways structure (Inducers → Intermediates → Effectors)
- layers: signaling pathways, functional modules, biological processes

Three cell-type specific maps (macrophages, NK, DC) can also be used separately. However, the combined meta-map contains more interactions and can often bring a complete picture.

To illustrate the use of the innate immune maps, we used the scRNA-seq of macrophages and NK cells from metastatic melanoma (section: “**Application of innate immune maps for high-throughput cancer data visualization and analysis**” of ?]).

We used **ICA** to identify factors driving diversity of the cells within the population of macrophages and NK single cells. We split NK and macrophages along the axis of the first independent component. Then the functional phenotypes of the subgroups of cells were analyzed with the innate immune map.

We selected from the single cell profiles the genes present in the innate immune map. For each functional map module, we computed an activity score which corresponds to the mean of 50% most variant genes between two groups. The activity scores were projected on the innate map facilitating the visual representation. The ensemble of the results allowed the further interpretation of functional phenotypes of NK and Macrophage groups.

For macrophages, we identified **pro- and anti-tumor polarisation**. The expression of inflammatory cytokines that induces local adaptive immunity via the antigen presentation process was the marker of the anti-tumor macrophage activity. The expression of immunosuppressive cytokines and growth factors supporting tumor growth characterized the pro-tumor phenotype.

Among NK cells we identified **tumor-killing** and **immunosuppressed phenotypes**. The upregulation of map modules: Lytic granules exocytosis, Recruitment of immune cells, Integrins, Fc receptors, Danger signal pathway were upregulated in the tumor killing phenotypes. We suggested that the tumor-killing would possibly be cells that are recently recruited and actively migrating. In contrast, an immunosuppressed group of cell seems to be the resting group that does not express strongly any anti-tumor activity.

We also identified possible molecular pathways differentially regulated between the phenotypes (Fig 5D ?]).

Besides, we demonstrate that the genes present in the map are significantly linked with the prediction of patients survival (both good and bad prognosis) therefore the map could be a support to analyze patient samples with conclusions sensibly affecting the prognosis.

8.3 Discussion and perspectives

In this work, we constructed the cell-specific and the meta innate-immune map including intra- and intercellular signaling in the Tumor Microenvironment. Using ICA, we defined groups of cells of scRNA-seq, and we interpreted their functional phenotypes using the constructed map.

We presented here a quite simplistic view of “heterogeneity” focusing on two groups for each cell type. It cannot illustrate the full complexity of the immune cell types interacting with the tumor. With increasing accessibility of single-cell data, it will be possible to perform multidimensional analysis to discover more functional subtypes that might also be dependent on tumor type, patient clinical features, and the treatment. Here we presented the first trial, shortly after the publication of the first cancer single cell RNAseq [?]. An interesting extension could be a comparison of NK and Macrophages from melanoma with the ones sequenced in other cancer since then, i.e., CRC or Brest cancer. It could also be interesting to project data from different patients on the complete ACSN 2.0 (cancer cell and TME), using different cell types: cancer, immune, etc., under

a condition, that they would have equilibrated and a sufficient number of cells sequenced per patient, to see possible differences between patients.

In the context of the thesis, this work demonstrated that the deconvolution of single-cell transcriptomes is possible and useful. It represents a deeper *zoom in* level if contrasted with the bulk transcriptome deconvolution performed so far. It demonstrates that ICA (and probably other deconvolution techniques) can also be used with single cell profiles to obtain coarse grain dissection of functional cell space.

A multiscale signalling network map of innate immune response in cancer reveals signatures of cell heterogeneity and functional polarization

Maria Kondratova^{1,3}, Urszula Czerwinska^{1,3,5}, Nicolas Sompairac¹, Sebastian D Amigorena², Vassili Soumelis², Emmanuel Barillot¹, Andrei Zinov'yev^{1,4} and Inna Kuperstein^{1,4*}

¹Institut Curie, PSL Research University, Mines Paris Tech, Inserm, U900, F-75005, Paris, France

²Institut Curie, PSL Research University, Inserm, U932, F-75005, Paris, France

³Equal contribution

⁴Senior authors

⁵Université Paris Descartes, Centre de Recherches Interdisciplinaires, Paris, France

*Correspondence: inna.kuperstein@curie.fr

SUMMARY

Lack of integrated resources depicting the complexity of innate immune response in cancer represents a bottleneck in the integration and interpretation of high-throughput data. To address this challenge, we performed systematic manual literature mining of molecular mechanisms governing innate immune response in cancer and represented it as an integrated signalling network map, where the knowledge was organised in a hierarchical and multiscale manner.

First, the individual cell-type specific signalling maps were constructed for macrophages, dendritic cells, myeloid-derived suppressor cells and natural killers. The cell type-specific maps were integrated into a comprehensive meta-map of innate immune response in cancer, depicting functional modules collectively contributing to anti- and pro-tumor signalling. Intuitive map exploration is enabled through our Google Maps-based NaviCell web platform. The meta map contains 1466 nodes among which there are 582 proteins, 1084 biochemical reactions and it is supported by information from 837 cell type specific and cancer-relates articles.

The cell type-specific signalling maps together with the meta-map of innate immune response in cancer form an open web-based platform. This unique resource allows deciphering the heterogeneity of cell populations in the tumor microenvironment (TME). All the developed maps are available online (<http://navicell.curie.fr/pages/maps.html>). It can be applied for innate immune status inference and making prognosis from based on tumor molecular profiling

The cell type-specific maps and the meta-map were used to interpret single cell RNA-Seq data from macrophages and natural killer (NK) cells in metastatic melanoma. The analysis demonstrated existence of sub-populations within each cell type that possess anti- and pro-tumor polarization status. Macrophage population consists of two types of cells, one is characterized by anti-tumor activity, whereas the second one is oriented towards pro-tumor activity. Activated subset of NK cells were characterized by induction of LFA1, CR3 and FcGR2 pathways involved in triggering tumor-killing signaling, indicating anti-tumor polarization status of NK in the studied sub-sets of cells.

Key words

Tumor immunology, tumor microenvironment, innate immunity signalling, cancer systems biology, comprehensive signalling network map, semantic zooming, single cell data analysis, bioinformatics, molecular pathways and networks, intercellular communication, cell reprogramming, polarization, heterogeneity

INTRODUCTION

Tumors are engulfed in a complex microenvironment (TME) that critically impacts disease progression and response to therapy. TME includes immune and non-immune interconnected components exchanging multiple signals and influenced by molecules secreted by cancer cells. The behavior of the tumor and its TME as a whole critically depends on the organization of these different players and their ability to regulate each other in a dynamic manner (Becht et al., 2016). The innate immune part of TME plays important, but sometimes opposite roles in tumor evolution. Innate immune cells can contribute to the elimination of tumor, e.g. through phagocytosis and T cell priming and by induction of adaptive immune response. However, they can also favor tumor escape from immunological control by a production of immunosuppressive molecules such as TGFB, IL10 and growth factors (Calì et al., 2017). An additional level of complexity in the TME is that various stimuli can lead to a range of innate immune cells phenotypes. This results in very heterogeneous sub-populations within each innate immune cell type coexisting in TME (Laoui et al., 2011), (Van Overmeire et al., 2014)(Chávez-Galán et al., 2015).

Depending on the set of stimuli from TME and tumor, immune cells are able to change their phenotype or polarization status from anti-tumor or pro-tumor (Vesely et al., 2011). Such functional dichotomy was first evidenced for one of the components of innate immunity in TME, the tumor-associated macrophages (TAM) and led to a description of M1 and M2 polarized TAM classes (Goswami et al., 2017). The same tendency was later documented for other components of innate immunity as Neutrophils (Fridlender and Albelda, 2012), Dendritic Cells (Gordon et al., 2014) and Natural Killers (Cooper et al., 2001). Therefore, the term ‘polarization’ can be applied for the innate immunity system in TME in general (Mittal et al., 2014) that represents the major interest of current works. The balance between anti-tumor and pro-tumor activity of innate immune cells has an impact on tumor growth, patient response to therapy and survival (Marvel and Gabrilovich, 2015).

Correct evaluation of the polarization status within the subtle innate immune cell sub-populations in TME is essential for immunotherapy improvement. Immune checkpoint targeting significantly changed the place of immunotherapy in cancer care. The most known immune therapeutic targets are PD-1/PD-L1 and CTLA-4, which inhibit adaptive (T cell) responses in tumors (Topalian et al., 2015). Currently, the majority of efforts are still invested into an identification of additional immune targets in the adaptive immune system (Torphy et al., 2017). However, the primary activation of adaptive immune response requires innate immune players, the antigen presenting cells (APC) such as dendritic cells (Vo et al., 2017) or macrophages (Mantovani et al., 2017)(Bonelli et al., 2017). Therefore, an efficiency of immune checkpoint therapy is directly dependent on proper innate immune activation (Moynihan and Irvine, 2017). In addition, there are studies showing that innate immunity can restrict tumor growth even when the adaptive immune system is

inactivated (O’Sullivan et al., 2012). This indicates that detailed study of potential innate immune-related targets should be performed to identify new types of immunotherapy (Tokunaga et al., 2018) that could function in synergy with the current T cell-targeted therapies or act independently (Bellora et al., 2017)(Gebremeskel et al., 2017).

There is a massive information in the literature about molecular mechanisms implicated in innate immune cells polarization in TME. However, most of the studies are focused on individual molecular components and pathways. They do not integrate the complexity of multiple crosstalk between innate immune cells and tumor. Similarly, there exists a number of immune-related gene signatures in different cancer types and stages, serving as immune status biomarkers and prognostic factors (Ascierto et al., 2011; Clark, 2017; Garg et al., 2017). However, the signatures do not explain the molecular mechanisms of polarization of innate immune cells in TME. To create a holistic picture of diversity and integrity of innate immune system in TME, the knowledge about molecular circuits should be gathered together and systematically represented (Kreuzinger et al., 2017).

To address these challenges, a systems biology approach is needed (Bhinder and Elemento, 2017). Formalization of biological knowledge in a form of comprehensive signalling maps, both at the intra- and intercellular levels helps to integrate information from multiple research papers (Dorel et al., 2015). Despite the fact that there are numerous public databases containing signalling pathways related to innate-immune response as KEGG (Kanehisa et al., 2012) and REACTOME (Croft et al., 2014), the signalling is represented there in patched manner lacking cross-regulatory links between pathways and integrated presentation of multi-cellular system of innate immunity. In addition, there are resources dedicated to different types of innate immune cells such as macrophages (Raza et al., 2008) or dendritic cells (Cavalieri et al., 2010). Finally, there are resources depicting the innate immune system in general as Innate DB (Breuer et al., 2013) and ImmuNet (Gorenshteyn et al., 2015), Virtually Immune (O’Hara et al., 2016). However, these repositories are rather pathogen response-oriented than cancer-specific and often represent a catalogue of disconnected pathways. Therefore, there is a need to create an integrated resource on molecular mechanisms of innate immune response in cancer. To fill the gap, we constructed a system of cell type-specific maps and an integrated meta-map of innate immune signalling in cancer based on the information retrieved from the literature (Figure 1). These maps together represent an open source analytic platform for data interpretation and modelling of TME in cancer and other human diseases.

RESULTS

Principles of innate immunity map construction and annotation

To cope with a massive body of literature on innate immune response in cancer we followed a systematic procedure of literature selection, knowledge organisation and integration of information in a visual and understandable manner (Figure 1). The network map is constructed as a two-dimensional map to facilitate a graphical representation of molecular mechanisms that drive biological processes. The map normally possesses a particular layout that reflects the accepted vision of spatial organisation and propagation of biological processes. Molecular mechanisms regulating six innate immune cell types found in the TME are gathered and depicted in the form of network maps. The information

about molecular mechanisms was manually retrieved by the map managers from the scientific literature along with the information presented in general pathway databases or in the immune system-specialized resources. The information was classified by specificity to the innate immune cell-types in cancer and organized into three cell-types specific signalling network maps, namely map of macrophages and myeloid-derived suppressor cells, dendritic cells and natural killer cells (Figure 2). These maps, enriched by the information on additional cell types as neutrophils and mast cells, were integrated into the meta-map of innate immune response in cancer (Figure 3).

The molecular mechanisms are depicted on the maps in the form of biochemical reaction network using well-established methodology (Kondratova et al., 2016)(Kuperstein et al., 2015). The maps are constructed using Systems Biology Graphical Notation language (SBGN) (Le Novère et al., 2009) as drawn in CellDesigner tool (Kitano et al., 2005) that ensures compatibility of the maps with various tools for network analysis, data integration and network modelling (Figure 3B). Each molecular player and reaction in the maps is annotated in the NaviCell format. The NaviCell annotations include PubMed references, cross-references with other molecular biology databases and notes of the map manager. In addition, each molecular player and reaction is assigned with the confidence score and tags that indicate involvement in different biological processes on the map (Supplemental Figure 1).

The principles and procedure of map construction, including the graphical standard, data model, rules of literature curation, data input from other databases and the detailed tagging system description are provided in the Methods section. In the future, the cell type-specific and the meta-map of innate immune response will become a part of the Atlas of Cancer Signalling Network resource (ACSN, <http://acsn.curie.fr>) and a tool for interactive web-based data visualization (Kuperstein et al., 2015).

Content and structure of the maps: inter- and intra-cellular signalling

Cell type-specific maps

The most studied and comprehensively described cell types in TME as macrophages and MDSC, dendritic cells and natural killers are represented in a form of individual cell type-specific maps. The correspondence of the mechanisms to anti- or pro-tumor activity for each cell type is indicated (Figure 2 and Supplemental Table 1). However, neutrophils and mast cells are less studied and molecular mechanisms implicated in the regulation of these cell types in TME is limited. The available knowledge on the neutrophils and mast cells in TME is included only into the meta-map of innate immune response in cancer (Figure 3 and Table 1).

Macrophages and myeloid-derived suppressor cells in cancer

Macrophages are the major immune component of leucocyte infiltration in the tumor. The main markers of macrophage are CD11b+, CD68+, LGALS+ and CD163. The anti-tumor polarization of macrophages is related to their ability to recognize and to reject tumor cells by phagocytosis, represent tumor antigens on the cell surface and induce a T-cell response. Macrophages produce cytotoxic agents such as Reactive Oxygen Species and Nitrite Oxide, secrete chemokines as CXCL-8, CCL2, CCL3, etc. and attract immune cells into the TME. It addition, they express inflammatory cytokines as TNF, IL12, IL1, etc. facilitating local immunity activation. Tumor-associated macrophages (TAMs) can also act as pro-tumor agents, expressing tumor stimulating growth factors as PDGF, EGF, VEGF, FGF, producing immunosuppressive molecules as IL10 and TGFB, that induce angiogenesis and matrix remodelling in TME

and consequently facilitate metastatic process (Biswas and Mantovani, 2010; Murray and Wynn, 2011)

Myeloid-derived suppressor cells (MDSC) represent a heterogeneous population of myeloid cells. In general, the role of MDSC in TME is similar to TAMs. Their main surface markers are CD33+, CD15+ (granulocytic), CD14+ (monocytic), CD34+ and CD11b+. MDSC suppress T-cell response and Natural killers' activity via TGFB signalling and arginine depletion from TME. In addition, MDSCs induce EMT and angiogenesis and participates in matrix remodelling via VEGF and MMPs secretion. MDSC mostly show a pro-tumor activity, therefore their presence in the tumor is correlated with a poor clinical prognosis (Gabrilovich and Nagaraj, 2009; Ostrand-Rosenberg and Sinha, 2009). The MDSC signalling is included into the Macrophage cell type-specific map.

The macrophage and MDSC cell type-specific map contains 588 objects and 7 modules represented both pro-tumor and anti-tumor polarization of myeloid cells. Pro-tumor zone includes module "Immunosuppressive cytokine pathways". Anti-tumor zone contains modules: "Immunostimulatory cytokine pathways", "Immunostimulatory cytokine expression", "Antigen presentation", "No and ROS production". Modules "Core signalling pathways" and "Recruitment of immune cells" form a natural border between pro- and anti-tumor zones (Figure 2A and Supplemental Table 1). and the map is available at http://navicell.curie.fr/pages/maps_macrophage.html.

Dendritic cells in cancer

Dendritic cells (DC) are innate immune cells that can have both myeloid and lymphoid origin. The main marker for mature dendritic cells is CD83+. Immature dendritic cells express among others HLA-DR, CD80, CD86, CD1a, CD40, CD14, CD11c, CD209, ILT3. As with macrophages, dendritic cells possess phagocytic abilities and can produce inflammatory cytokines as IFNs, IL12, etc. The major role of dendritic cells in anti-tumor response is antigen presentation and T-cell activation. (Palucka and Banchereau, 2012) . The DC map contains 491 objects and 8 modules (Figure 2B and Supplemental Table 1) Pro-tumor zone of the map contains modules "Immunosuppressive cytokine pathways", "Immunosuppressive checkpoints". Anti-tumor zone contains modules: "Immunostimulatory cytokine pathways", "Antigen presentation", "Tumor recognition and tumor killing", "DC markers", modules "Core signalling pathways" and "Recruitment of immune cells" form a border between pro- and anti-tumor zones.

The map is available at http://navicell.curie.fr/pages/maps_dendritic.html.

Natural killers cells in cancer

Natural killers (NK) are big granular lymphocytes which can be cytotoxic to tumor cells. The markers of this cell type are specific NK-receptors as NKp30, NKp46, NKG2D, etc. The main role of NK cells in innate immunity is an elimination of cells lacking MHC1 molecules that therefore cannot be recognized by T-cells. NK are stimulated by the target cells expressing NK receptors activating ligands such as MICA, MICB, etc. The activity of NK cells is modulated by inflammatory cytokines as IL15, IL12, produced by macrophages and dendritic cells. NK cells secrete granules contains lytic enzymes (granzymes, perforin, granzulin, etc) and express apoptosis inducers TRAIL and FASL. Presence of active NK cells in cancer is correlated with good prognosis. To escape NK control, tumor cells express immunosuppressive ligands as MIF, IL10, TGFB, and downregulation of NK ligands expression that collectively inhibit cytotoxic activity of NK cells (Vivier et al., 2012). A pro-tumor polarization of NK cells is not described in the literature. However, suppressed NK cells are incapable to reject tumor cells and therefore indirectly

promote cancer progression. The NK map contains 567 objects and 6 modules (Figure 2C and Supplemental Table 1). Pro-tumor zone of the map contains modules “Immunosuppressive cytokine pathways”, “NK inhibiting receptors”. Anti-tumor zone contains modules: “Immunostimulatory cytokine pathways”, “NK activating receptors”, “Lytic granules exocytosis”, module “Core signalling pathways” forms a border between pro- and anti-tumor zones.

It is available at http://navicell.curie.fr/pages/maps_natkiller.html.

Neutrophil cells in cancer

Neutrophils form a subtype of granulocytic leukocytes. The main markers of this cell type are (FUT4, CD16, ITGA4(-)). The role of neutrophils in the tumor microenvironment is not well documented, but it is known that they can produce ROS, inflammatory cytokines and demonstrate tumoricidal activity. Though in other conditions neutrophils act as pro-tumor agents via stimulation of matrix remodelling, angiogenesis and metastasis, therefore these cells have both pro and-antitumor polarization potential (Fridlender and Albelda, 2012; Fridlender et al., 2009) . The signalling on neutrophils is included into the innate immune meta-map (Figure 3 and Table 1).

Mast cells in cancer

Mast cells resemble blood basophils and contain granules rich in histamine and heparin (markers: FCER2, KIT, ENPP3, FCER1A). Experimental data about the influence of mast cell on tumor microenvironment is contradictory. It is known that mast cells can produce inflammatory cytokines IL1, IL6, TNF, INF α and secrete Chondroitin sulphate which acts as a decoy for tumor cells and blocks the metastatic process. However, mast cells also secrete molecules stimulating tumor growth, angiogenesis and local immunosuppression (Tryptase, Heparin, IL8, VEGF, NGF, PDGF, SCF, Histamine) (Marichal et al., 2013; Theoharides and Conti, 2004) . Probably the polarization of mast cells in TME is context-dependent. The signalling on mast cells is included into the innate immune meta-map (Figure 3 and Table 1).

Integrated hierarchical modular meta-map of innate immune response in cancer

The aforementioned cell type-specific maps gathered together and enriched by additional information, gave rise to the global, seamless meta-map of innate immunity in cancer. The layout design of the meta-map reflects the current understanding of signalling propagation in cells. To cope with the complexity of the signalling network and to make it understandable and navigable, the meta-map has a hierarchical structure (Figure 1 and Figure 3). The meta-map possesses two major structuring dimensions: the internal organisation of the map (layers, zones, modules, pathways) and external organisation represented by zoom levels (see the explanation below).

The internal organisation of the meta-map is provided in a form of *three layers*: Inducers, Core signalling and Effectors (Figure 3B and Table 1). The top part of the meta-map depicts inducer molecules frequently present in TME (layer ‘Inducers’). The inducers interact through specific receptors and adaptor proteins that propagate the signal via limited number of transmitters, also called hub molecules as NF- κ B, PLCG, PI3K etc. These molecules are located in the middle parts of the meta-map in the layer Core signalling. The signalling is further propagated to the layer Effectors which are located in the lower part of the meta-map. The latest entities actually execute the biological activity and therefore define the outcome phenotype, namely, the positive or negative influence of the innate immunity system on

the tumor growth and invasion (Figure 3B and Table 1).

Further, the whole meta-map is divided into multiple *signalling pathways*, running through the aforementioned layers. (Figure 3B) We define a signalling pathway as a natural sequence of molecular interactions which transform extracellular signal into intracellular activity. The pathways include all macromolecules as proteins, RNA, genes, etc. participating or influenced by a certain ligand or receptor and leading to a particular cell outcome indicated by phenotypes such as Tumor Killing, Tumor Growth, etc. Usually, pathways are named after the first molecule in the sequence, which is a ligand (e.g. TNF pathway, IL10 pathway), but in the cases when several ligands act through the same receptor, a signalling pathway receives the name of the corresponding receptor-ligand complex (e.g. TLR2/4 pathway). The meta-map is composed of 98 signalling pathways, 30 of which contain more than 10 molecules in the sequence (Supplemental Table 2). It is worth highlighting that there are many cross-talks between different signalling pathways (Figure 3B). The signalling pathways on the meta-map are useful for retrieving the back-bone structure of the network and map reduction, especially relevant for structural analysis and modelling studies.

The signalling pathways of the meta-map form together 25 *functional modules* representing relatively independent fragments of global network responsible for execution of certain molecular functions, e.g. Antigen presentation, Exocytosis, Phagocytosis, Checkpoints, etc. The functional modules are assembled into the structures of higher level, namely 9 *biological processes (meta-modules)*, reflecting the major biological activities of innate immune system with respect to a tumor, i.e. Tumor recognition, Tumor growth, Timor killing, Immune stimulation, Immune suppression, etc. At the highest level, all biological processes are grouped into two *zones* representing the concept of innate immune system polarization into anti- or pro-tumor mode. The anti-tumor zone contains ‘Tumor recognition’, ‘Immune activation’, ‘Tumor killing’ processes, whereas the pro-tumor zone is composed of ‘Inhibition of tumor recognition’, ‘Immune suppression’ and ‘Tumor growth’ processes (Figures 1, 3A and Table 1). The list of molecules per modules and biological processes (meta-modules) is available in the Supplemental Table 3. All these levels of the map are interconnected and cross-talk to each other. The cross-talks between key biological processes is represented as an interaction network (Figure 3C) We can see a number of interactions between different parts of the map here, of different nature (activation, inhibition, molecular flow) and different scale. We can see that “Core signalling” (central point of network) play a role of the “hub” for most signalling pathways and that there are a lot of positive and negative cross-talks between immune stimulation and immune suppression in innate immunity.

The meta-map contains 1466 nodes among which there are 582 proteins, 1084 biochemical reactions and it is supported by information from 820 cell-type specific and cancer-relates articles (Table 1). The meta-map and cell specific maps are available at http://navicell.curie.fr/pages/maps_innateimmune.html.

The external organisation of the meta-map is reflected in *hierarchical structure of zoom levels*, similar to geographical maps, where on each zoom level only limited information is displayed. The meta-map contains the top zoom level, the least detailed level that schematically represent borders of Biological Processes, providing the global view on the map organisation. The contours of the meta-modules and modules are highlighted by the colourful background (Figure 3A). The next zoom level allows to appreciate the functional modules structure and the last, most detailed one provides the view on signalling pathways and detailed biochemical reactions that compose these pathways (Figure 3 B). This hierarchical structure of the map facilitates Google Maps-like navigation of the map as explained in the next section.

Access, navigation and maintenance of the innate immune response maps

The cell type-specific and the integrated meta-map are open source and can be browsed online. The user-friendly interface of the maps provides a possibility to explore the individual cell type-specific maps or to access the integrated meta-map of innate immunity in cancer. The visualization and navigation of maps are supported by the NaviCell web-based environment empowered by Google Maps engine (Kuperstein et al., 2013). The navigation features such as search, scrolling, zooming, markers, callout windows and zoom bar are adopted from the Google Maps interface. All map components are clickable, making the map interactive. The extended annotations of map components contain rich tagging system, converted to links. This allows tracing the involvement of molecules into different map sub-structures as pathways, modules, and biological processes. In addition, cell type-specific tags make clear correspondence between the cell type-specific maps and the integrated meta-map of innate immune response, allowing shuttling between the maps (Supplemental Figure 3).

The semantic zooming feature of NaviCell (Kuperstein et al., 2013) simplifies navigation through large maps of molecular interactions, showing readable amount of details at each zoom level. Gradual exclusion of details allows exploration of map content from the detailed towards the top-level view. The hierarchical structure of the innate immune response meta-map as described above, allowed to generate several zoom levels (Figure 3 A and B).

Comparison of meta-map of innate immune response in cancer with existing pathway databases

The content of the meta-map was compared with the relevant sub-set of pathways related to the innate immune system from existing molecular interaction databases. The InnateDB database contains a detailed description of the innate-immune signalling, even though more general databases as KEGG and REACTOME also include immune pathways. Pathways related to the human innate immune system were selected from the InnateDB resource, excepting ‘Complement Cascade (Human)’, ‘NOD-like Receptor Signalling Pathway’, ‘Regulation of autophagy (Human)’, ‘RIG-I-like receptor signalling pathway (Human)’. The excluded pathways represent virus and bacterial infection-specific pathways that do not correspond to TME signalling. The KEGG innate immune-related pathways were retrieved from the list “5.1 Immune system”. The pathways obtained from REACTOME cover ‘Class I MHC mediated antigen processing & presentation’ and ‘MHC class II antigen presentation’ from Adaptive Immune branch, and all pathways from Innate Immune branch (the list of selected pathways is available in the Supplemental Table 4). All together 666 gene names from Innate DB, 563 gene names from KEGG and 2156 gene names from REACTOME were selected and compared with the innate immune response meta-map that contains 683 gene names.

The selected InnateDB pathways altogether contain nearly the same number of objects as the innate immune response meta-map. The content of selected KEGG or REACTOME pathways is richer than in the innate immune response meta-map, due to the fact that KEGG and REACTOME are generic databases, describing all innate immune-related interactions, whereas the meta-maps is rather oriented to cancer signalling. The overlap between the meta-map and the three selected databases represents 61 % for InnateDB, 58% for KEGG and 30% for REACTOME. It is important to note that there are 188 genes that present exclusively at the innate immune response meta-map (Supplemental Figure 3A and Supplemental Table 4). These unique genes are relatively homogeneously distributed across the meta-map, indicating that depicted processes are described in more depth on the meta-map compared to other three databases (Supplemental Figure 3 A). There are several modules that significantly enriched by unique genes on the meta-map

(Supplemental Figure 3 B). Thus, the modules ‘Tumor growth’ and ‘Immunosuppressive checkpoints’ contain signalling that very well studied in cancer cells and therefore represented in great details on the meta-map. There are additional two modules, entitled ‘MIRNA TF immunostimulatory’ and ‘MIRNA TF immunosuppressive’, that contain the latest information on involvement of miRNA in the innate immune system control in cancer and unique for the meta-map, comparing to other databases. We concluded that the content of meta-map is relatively non-redundant with the other pathway databases and there are several functional modules directly related to TME functions, that are unique to the meta-map.

We also compared the set of publications used to annotate the four pathway databases. The overlap of the literature body in the meta-map with the three databases was small: 785 papers out of 837 papers that were used to annotate the meta-map are unique (Supplemental Figure 3B). Although the median date of annotated reference in the meta-map is only one year-younger compared with InnateDB and REACTOME, there is significant number of papers dating 2010–2017 (230 Articles (27%) out of 837). The meta-map contains more papers published after 2010 than Innate DB and REACTOME (Supplemental Figure 3C), indicating that the map contains the most recent discoveries in the corresponding fields. The meta-map uses relatively similar range of journals as the other two databases, however the specific immunological journals (such as J. of Immunology, Immunity, Nat. Immunology) and cancer-specific journals (such as Cancer Res. and Oncogene) are used much more frequently comparing to the other two databases. The two other databases are rather oriented towards more generic molecular biology journals as JBC, MCB, Nature and PNAS (Supplemental Figure 3 C, D). We further compared the features of innate immune response representation in different pathway databases. Our innate immune response in cancer resource is the one that contains cell type-specific maps in opposite to other databases. The comparison indicates that the cross talk between the pathways is virtuously represented at the maps of immune response in cancer resource. Finally, the combination of hierarchical organization of knowledge and possibility of navigation through the layers of the maps dues to semantic zooming feature makes the innate immune resource more suitable for meaningful data visualisation (Table 3). The visualisation tool box is build-in into the NaviCell environment that allows easy data integration and visualisation in the context of the innate immune maps. Taking together, the results of databases comparison indicate that the innate immune response in cancer resource is topic-specific, that describes immune-related and cancer-relevant signalling processes based on the latest publications about innate immune component in TME. The thoughtful layout and visual organisation of the biological knowledge on the maps makes it a distinguished resource for data analysis and interpretation.

Application of innate immune maps for high-throughput cancer data visualization and analysis

The cell type-specific maps and the meta-map were applied to explore the heterogeneity of innate immune cell types in cancer. The single-cell RNA-Seq data for macrophages and natural killer (NK) cells from metastatic melanoma samples were used (Tirosh et al., 2016).

Polarization and heterogeneity of macrophages population in melanoma

With the help of unsupervised independent component analysis (ICA)-based methodology of gene expression analysis (Hyvärinen and Oja, 2000), we decomposed single cell transcriptome data of Macrophage cells into independent factors. When the single cell RNASeq profiles of individual macrophages were projected in a two-dimensional space (Principal Component 1 and 2), one can see that the independent component computed are attracted by the bimodality

characterizing the distribution (Figure 4A). In order to functionally characterize the biological factor driving this bimodality, data points from the extreme opposite sides of the independent component direction were selected, defining Groups 1 and 2 respectively (see STAR Methods). Then we analyzed potential pro- and anti-tumor properties of these Macrophage cell groups in the context of the innate immunity meta-map. Group 1 has significantly higher anti-tumor score (t-test p-value: 0.02) and Group 2 is the pro-tumor one (t-test p-value: 0.003).

The expression profile differences of the cells from the two groups were interpreted in the context of the Macrophage cell type-specific map and the innate immune response meta-map. The results of the enrichment study for the two Macrophage groups were also represented as heatmaps with a significance level of the p-value for student t-test (see STAR Methods) (Supplemental Figure 4). The module activity values were plotted on the maps using BiNoM plugin of Cytoscape (Bonnet et al., 2013).

Visualization of the data in the context of macrophage cell type-specific demonstrates that the module ‘Antigen presentation’ is upregulated in Macrophage Group 1 (Figure 4B) comparing to Macrophage Group 2 (Figure 4C). Whereas, Macrophage Group 2 (Figure 4C) shows upregulated modules ‘Core signalling pathways’ and ‘Immunosuppressive cytokines pathways’ comparing to Macrophage Group 1 (Figure 4B).

Then, the expression data for the two Macrophage cell groups were analysed in the context of the meta-map that allowed to detect several additional modules differentially regulated between the two groups. The four modules ‘Antigen presentation’ ‘Immunosuppressive checkpoints’, ‘Danger signal modules’ and ‘Immunostimulatory MiRNA and TF’ were significantly overexpressed in Anti-tumor Macrophage Group 1 (t-test p-values respectively: $<10^{-4}$, 0.009, $<10^{-8}, <10^{-5}$, Figure 4C) comparing to Pro-tumor Macrophage Group 2 (Figure 4D and E). In contrary, the three modules ‘Recruitment of immune cells module’, ‘Tumor Growth’ and ‘Immunosuppressive cytokine expression’ were strongly upregulated in Pro-tumor Macrophage Group 2 (t-test p-values respectively: $<10^{-6}, <10^{-6}, <10^{-5}$, Figure 5D), in comparison to Anti-tumor Macrophage Group 1 (Figure 4 D and E).

From these results, it can be concluded that the Macrophage Group 1 has a tendency to express an *anti-tumor phenotype*, because it is characterized by expression of inflammatory cytokines that are able to induce local adaptive immunity via Antigen presentation process. Interestingly, the most typical modules responsible for tumor elimination as ‘Exocytosis and Phagocytosis’ and ‘Immunostimulatory cytokine pathways’ are not over-activated in this cell subset.

In contrary, Macrophage Group 2 demonstrate a *pro-tumor phenotype*, characterized by expression of both, immunosuppressive cytokines restricting local immune response and growth factors supporting tumor growth.

Polarization and heterogeneity of natural killer cells population in melanoma

After using as previously ICA decomposition to group samples, we computed the module activity scores of each group and then a t-test to evaluate the difference in module activity between two NK subpopulations (Group 1 referred to as “tumor-killing” and Group 2 referred to as “immunosuppressed”). (Figure 5A, Supplemental Figure 5A and Supplemental Figure 5B).

First, the comparison and visualisation of the module activity between the two NK cells groups demonstrated activation

of ‘Lytic granules exocytosis’ module in NK Group 1 (the ‘tumor-killing group’) (Figure 5B) comparing to NK Group 2 (the ‘immunosuppressed group’) (Figure 4B) (t-test p-value: 0.006), on the NK cell type-specific map. The activity of this module is directly responsible for tumor killing capacity of NK Group 1 cells that most probably exposes stronger anti-tumor abilities comparing to Group 2. Next, the two NK cells groups were analysed in the context of the meta-map that allowed to detect five differentially regulated modules between the two groups of NK cells. The four modules ‘Recruitment of immune cells’, ‘Integrins’, ‘Fc receptors’, ‘Danger signal pathway’ were significantly upregulated in the NK Group 1 comparing to the NK Group 2 (t-test p-values respectively: 0.0001, $<10^{-4}$, 0.004, $<10^{-5}$). In contrary, the module ‘Immunosuppressive MiRNA and TF’ was inhibited in the NK Group 1 comparing to the NK Group 2 (t-test p-value: 0.001). Finally, although the activity of ‘Phagocytosis and Exocytosis’ module is not significantly different between the two groups, this module is rather activated in the NK Group 1 comparing to the NK Group 2.

Collectively these results demonstrate that the NK Group 1 is characterised by upregulation of biological functions related to NK cell recruitment and activation, coinciding with upregulation of the mechanisms responsible for tumor killing. Thus, the NK Group 1 can be interpreted as newly-recruited, actively migrating NKs with strong anti-tumor polarization. In contrary, most probably, NK Group 2 contains resting or suppressed NK cells that do not expose a well-defined phenotype.

Then we decided to explore possible molecular mechanisms that would explain simultaneous activation of upstream map zones (modules ‘Recruitment of immune cells’, ‘Integrins’, ‘Fc receptors’, ‘Danger signal pathway’) and downstream effector modules ‘Phagocytosis and Exocytosis’ at the level of signalling pathways in NK Group 1. We have compared activation of 30 well annotated pathways on the meta-map (each containing more than 10 proteins) between “tumor-killing group” (Group 1) and “immunosuppressed group” (Group 2) and presented results as a heat map (Figure 5C). There are 7 differentially regulated pathways, 5 upregulated in Group 1 (LFA1, CR3, STING, 2B4, FcGR2) and 2 upregulated in Group 2 (IL13, IL18) (t-test p-values <0.05). Within pathways activated in the “tumor-killing group” (Group 1) there are three (LFA1, CR3 and Fc γ RII). The key players of the pathways are presented schematically in Figure 5D. It can be concluded that the meta-map described difference between NK subtypes both on the level of functional modules and signalling pathways.

Innate immune response meta-map as a source of patient survival signatures

To study whether the innate immune response meta-map can be used for assessment of processes contributing to patient survival, we used the list of genes which have correlation with prognosis of patient survival from (Gentles et al., 2015) (see STAR Methods). We first verified the presence of the genes correlating with patient survival from the above study on the innate immune response meta-map. We detected that out of 627 proteins and protein coding genes depicted on the meta-map, 295 are significantly (z-score p-value <0.05) correlated with the patient survival, that represents 47% of the map content (vs. 27% in whole genome study). The genes enriched in the meta-map can be divided into two groups, positively and negatively correlated with the patient survival, which confirms the observation that innate immune system can play a dual role in cancer disease. Interestingly that from the whole genome analysis in the original study by Gentles et al., 2015, it emerges that there is quasi equal proportion of positively and negatively-correlating genes. However, in the innate immune response meta-map, there is a strong predominance of genes with positive

influence on patient survival (Table 2).

In order to highlight biological functions on the innate immune response map associated to positive or negative patient survival, mean values of gene z-scores per meta-modules were calculated and visualized in the context of the innate immunity meta-map (see STAR Methods). As a general trend, the layers ‘Inducers’ and ‘Core signalling’ on the meta-map are more significantly correlated with patient survival, comparing to the layer ‘Effectors’. Further, the modules with biological functions related to anti-tumor activity as ‘Immune response stimulation’ and ‘Tumor recognition’, ‘Recruitment of immune cells’, etc. are positively correlated with patient survival. Interestingly that the module ‘Tumor killing’ is also positive correlated with the patient survival, though not reaching the statistical significance (Table 2 and Supplemental Figure 6). The minority of functional modules related to pro-tumor activity as ‘Tumor growth’, ‘Immunosuppressive core pathways’, ‘Immunosuppressive MiRNA and TF’ are negatively correlating with patient survival (Table 2 and Supplemental Figure 6). Described analysis demonstrates that the meta-map can serve for evaluation of innate immune response signatures associated with patient survival in cancer.

DISCUSSION

The tumor microenvironment (TME) is now recognized as a critical determinant of tumor development and response to therapy. Its study and pharmacological manipulation are hampered by its complexity and plasticity of cellular components. Systems biology approaches are well suited to address either or both of these difficulties. Therefore, systematization and formalization of molecular mechanisms regulating TME in general, and the innate immune component of the system in particular, are needed. This should include the dissection of multiple intracellular interactions, as well as crosstalks between different TME cell populations with tumor cells.

One of the challenges of cancer biology today is understanding the phenomena of tumor heterogeneity. It consists of two relatively independent parts. First, it is a heterogeneity of the tumor cells themselves, as a result of their clonal divergence or action of epigenetic mechanisms. Second, it is a natural heterogeneity of tumor microenvironment (TME). The last years’ discoveries have shown that understanding how the components of this multi-cellular TME system interact with each other is very important for effective drug design. Actually, the attempt for modulation of the interactions in the tumor microenvironment lies in the basis of new anti-cancer immune check-point therapy.

One of the obstacles hindering the progress in the field is a large number of disconnected experimental data that are not integrated to create a holistic picture. In order to gather together the dispersed scientific knowledge, we have built the set of comprehensive network maps of innate immune response in cancer.

Analysis of large amount of scientific information and search for optimal forms of its representation required development of new approaches for network map construction and annotation. Our first goal was to preserve the natural multidimensionality of the biological knowledge available for different cell type in the innate component of TME. Indeed, different cells types in innate immune system are studied from different angles. Some signalling pathways are described in detail for the macrophages and others for natural killer cells and so on. It is clear that the molecular knowledge described for one cell type cannot be always extrapolated to another. This motivated us to create two complementary representations of innate immune system in cancer, one in the form of cell type-specific maps and the

second, as an integrated meta-map of innate immune response in cancer. To be able to trace the correspondence of molecular entities and processes to a particular cell type, we introduced a system of cell type-specific tags, included in to the annotation of all entities on the maps.

Our second goal was to provide a complete, but not too controversial picture on the processes occurring in the TME. Generation of an integrated meta-map of innate immunity immediately exposed a problem of map complexity. We coped with the complexity problem by introducing the hierarchical structure into the integrated meta-map, respecting the biological functions. The general layout of the integrated meta-map is based on the idea of immune cells polarization in TME, reflected in the representation of both, pro-tumor and anti-tumor signalling mechanisms leading to the corresponding phenotypes and the signalling responsible for a switch in the polarization state. In accordance with the literature, all functional modules and meta-modules on the map are grouped into the pro-tumor and anti-tumor zones.

The modular hierarchical map structure and complex tagging system of maps entities facilitated generation of geographical-like easily browsable open source repository. Taking an advantage of NaviCell platform, that provides Google Maps-engine and map navigation features, the innate immune maps can be explored in an intuitive way, allowing shuttling between the cell type-specific map to the integrated meta-map.

NaviCell-based representation of the maps facilitates visualization of various types of omics data. Analysis of data in the context of both, cell type-specific and integrated maps can help in the formalization of biological hypotheses for the processes and interactions that are studied in some cell types, but unexplored in others. In addition, thanks to the rich system of tags, the maps content can be used as a source of knowledge-based gene signatures of innate immune cell type. Finally, hierarchical organization of the map provides a basis for structural network analysis, complexity reduction and eventual transformation of the map into executable mathematical models.

The resource of innate immune maps is useful for computing network-based molecular signatures of innate immune cells polarization. These signatures will help to characterize the overall status of the signalling dictating pro-tumor and anti-tumor states of TME in cell lines and tumoral samples. It will also help to stratify cancer patients according to the status of the TME and potentially predict patient survival and response to immunotherapies. In addition, the resource might potentially provide new immunotherapy targets, among innate immunity components of TME in tumor infiltrates. These targets can be complementary or synergistic to the well-known immune checkpoint inhibitors.

Construction of innate immune response map is the first step in the attempt to build a global network describing molecular interactions in TME. The next perspective is to represent the knowledge on adaptive immune response and non-immune components in the tumor environment, as fibroblasts and endothelial cells. The final goal is to build a complete map of signalling in cancer representing both intracellular interaction of tumor cells and each component in TME and the intra-cellular interactions, describing the coordination between the component of this multicellular system.

ACKNOWLEDGMENTS

We thank D. Rovera for help with network structure analysis, L. Cristobal Monraz Gomez for help with data visualization and Nicolas Sompairac for help with statistical analysis of maps content. This work has been funded by INSERM Plan Cancer N° BIO2014-08 COMET grant under ITMO Cancer BioSys program. This work received support from MASTODON program by CNRS (project APLIGOOGLE) and COLOSYS grant ANR-15-CMED-0001-04, provided by the Agence Nationale de la Recherche under the frame of ERACoSysMed-1, the ERA-Net for Systems Medicine in clinical research and medical practice. ITMO cancer (AVIESAN) provided 3-year PhD grant and foundation Bettencourt Schueller and Center for Interdisciplinary Research supported the training of the PhD student.

AUTHOR CONTRIBUTIONS

M.K. constructed signalling networks, performed data visualisation and wrote the paper; U.C. performed data analysis and enrichment calculations and wrote the paper; S.D.A. advised during the project and revised the paper; V.S. advised during the project and critically revised and restructured the paper; E.B. advised during the project and revised the paper; A.Z. supervised the data analysis, advised during the project and revised the paper; I.K. led the project and wrote the paper.

DECLARATION OF INTERESTS

The authors declare no competing interests.

REFERENCES

- Ascierto, M., Giorgi, V. De, Liu, Q., Bedognetti, D., Spivey, T.L., Murtas, D., Uccellini, L., Ayotte, B.D., Stroncek, D.F., Chouchane, L., et al. (2011). An immunologic portrait of cancer. *J. Transl. Med.* *9*, 146.
- Becht, E., Giraldo, N.A., Dieu-Nosjean, M.-C., Sautès-Fridman, C., and Fridman, W.H. (2016). Cancer immune contexture and immunotherapy. *Curr. Opin. Immunol.* *39*, 7–13.
- Bellora, F., Dondero, A., Corrias, M.V., Casu, B., Regis, S., Caliendo, F., Moretta, A., Cazzola, M., Elena, C., Vinti, L., et al. (2017). Imatinib and Nilotinib Off-Target Effects on Human NK Cells, Monocytes, and M2 Macrophages. *J. Immunol.* *199*, 1516–1525.
- Bhinder, B., and Elemento, O. (2017). Towards a better cancer precision medicine: Systems biology meets immunotherapy. *Curr. Opin. Syst. Biol.* *2*, 67–73.
- Biswas, S.K., and Mantovani, A. (2010). Macrophage plasticity and interaction with lymphocyte subsets: cancer as a paradigm. *Nat. Immunol.* *11*, 889–896.
- Bonelli, S., Geeraerts, X., Bolli, E., Keirsse, J., Kiss, M., Pombo Antunes, A.R., Van Damme, H., De Vlaminck, K., Movahedi, K., Laoui, D., et al. (2017). Beyond the M-CSF receptor - novel therapeutic targets in tumor-associated macrophages. *FEBS J.*
- Bonnet, E., Calzone, L., Rovera, D., Stoll, G., Barillot, E., and Zinovyev, A. (2013). BiNoM 2.0, a Cytoscape plugin

for accessing and analyzing pathways using standard systems biology formats. *BMC Syst. Biol.* *7*.

Breuer, K., Foroushani, A.K., Laird, M.R., Chen, C., Sribnaia, A., Lo, R., Winsor, G.L., Hancock, R.E.W., Brinkman, F.S.L., and Lynn, D.J. (2013). InnateDB: systems biology of innate immunity and beyond—recent updates and continuing curation. *Nucleic Acids Res.* *41*, D1228–D1233.

Calì, B., Molon, B., and Viola, A. (2017). Tuning cancer fate: the unremitting role of host immunity. *Open Biol.* *7*, 170006.

Cavalieri, D., Rivero, D., Beltrame, L., Buschow, S.I., Calura, E., Rizzetto, L., Gessani, S., Gauzzi, M.C., Reith, W., Baur, A., et al. (2010). DC-ATLAS: a systems biology resource to dissect receptor specific signal transduction in dendritic cells. *Immunome Res.* *6*, 10.

Chávez-Galán L., Olleros, M.L., Vesin, D., and Garcia, I. (2015). Much More than M1 and M2 Macrophages, There are also CD169+ and TCR+ Macrophages. *Front. Immunol.* *6*, 263.

Clark, D.P. (2017). Biomarkers for immune checkpoint inhibitors: The importance of tumor topography and the challenges to cytopathology. *Cancer Cytopathol.*

Cooper, M.A., Fehniger, T.A., and Caligiuri, M.A. (2001). The biology of human natural killer-cell subsets. *Trends Immunol.* *22*, 633–640.

Croft, D., Mundo, A.F., Haw, R., Milacic, M., Weiser, J., Wu, G., Caudy, M., Garapati, P., Gillespie, M., Kamdar, M.R., et al. (2014). The Reactome pathway knowledgebase. *Nucleic Acids Res.* *42*, D472-7.

Dorel, M., Barillot, E., Zinovyev, A., and Kuperstein, I. (2015). Network-based approaches for drug response prediction and targeted therapy development in cancer. *Biochem. Biophys. Res. Commun.* *464*, 386–391.

Fridlender, Z.G., and Albelda, S.M. (2012). Tumor-associated neutrophils: friend or foe? *Carcinogenesis* *33*, 949–955.

Fridlender, Z.G., Sun, J., Kim, S., Kapoor, V., Cheng, G., Ling, L., Worthen, G.S., and Albelda, S.M. (2009). Polarization of tumor-associated neutrophil phenotype by TGF-beta: “N1” versus “N2” TAN. *Cancer Cell* *16*, 183–194.

Gabrilovich, D.I., and Nagaraj, S. (2009). Myeloid-derived suppressor cells as regulators of the immune system. *Nat. Rev. Immunol.* *9*, 162–174.

Garg, A.D., More, S., Rufo, N., Mece, O., Sassano, M.L., Agostinis, P., Zitvogel, L., Kroemer, G., and Galluzzi, L. (2017). Trial watch: Immunogenic cell death induction by anticancer chemotherapeutics. *Oncoimmunology* *6*, e1386829.

Gebremeskel, S., Lobert, L., Tanner, K., Walker, B., Oliphant, T., Clarke, L.E., Dellaire, G., and Johnston, B. (2017). Natural Killer T-cell Immunotherapy in Combination with Chemotherapy-Induced Immunogenic Cell Death Targets Metastatic Breast Cancer. *Cancer Immunol. Res.* *5*, 1086–1097.

Gentles, A.J., Newman, A.M., Liu, C.L., Bratman, S. V, Feng, W., Kim, D., Nair, V.S., Xu, Y., Khuong, A., Hoang,

C.D., et al. (2015). The prognostic landscape of genes and infiltrating immune cells across human cancers. *Nat. Med.* *21*, 938–945.

Gordon, J.R., Ma, Y., Churchman, L., Gordon, S.A., and Dawicki, W. (2014). Regulatory dendritic cells for immunotherapy in immunologic diseases. *Front. Immunol.* *5*, 7.

Gorenshteyn, D., Zaslavsky, E., Fribourg, M., Park, C.Y., Wong, A.K., Tadych, A., Hartmann, B.M., Albrecht, R.A., García-Sastre, A., Kleinsteiner, S.H., et al. (2015). Interactive Big Data Resource to Elucidate Human Immune Pathways and Diseases. *Immunity* *43*, 605–614.

Goswami, K.K., Ghosh, T., Ghosh, S., Sarkar, M., Bose, A., and Baral, R. (2017). Tumor promoting role of anti-tumor macrophages in tumor microenvironment. *Cell. Immunol.* *316*, 1–10.

Himberg, J., and Hyvärinen, A. (2003). ICASSO: Software for investigating the reliability of ICA estimates by clustering and visualization. In *Neural Networks for Signal Processing - Proceedings of the IEEE Workshop*, pp. 259–268.

Hyvärinen, A., and Oja, E. (2000). Independent Component Analysis: Algorithms and Applications. *Neural Networks* *13*, 411–430.

Kanehisa, M., Goto, S., Sato, Y., Furumichi, M., and Tanabe, M. (2012). KEGG for integration and interpretation of large-scale molecular data sets. *Nucleic Acids Res.* *40*, D109-14.

Kitano, H., Funahashi, A., Matsuoka, Y., and Oda, K. (2005). Using process diagrams for the graphical representation of biological networks. *Nat. Biotechnol.* *23*, 961–966.

Kolde, R. (2012). Package ‘pheatmap’. *Bioconductor* 1–6.

Kondratova, M., Barillot, E., Zinovyev, A., and Kuperstein, I. (2016). Knowledge Formalization and High-Throughput Data Visualization Using Signaling Network Maps. *bioRxiv* 089409.

Kreuzinger, C., Geroldinger, A., Smeets, D., Braicu, E.I., Sehouli, J., Koller, J., Wolf, A., Darb-Esfahani, S., Joehrens, K., Vergote, I., et al. (2017). A Complex Network of Tumor Microenvironment in Human High-Grade Serous Ovarian Cancer. *Clin. Cancer Res.*

Kuperstein, I., Cohen, D.P.A., Pook, S., Viara, E., Calzone, L., Barillot, E., and Zinovyev, A. (2013). NaviCell: a web-based environment for navigation, curation and maintenance of large molecular interaction maps. *BMC Syst. Biol.* *7*, 100.

Kuperstein, I., Bonnet, E., Nguyen, H.-A., Cohen, D., Viara, E., Grieco, L., Fourquet, S., Calzone, L., Russo, C., Kondratova, M., et al. (2015). Atlas of Cancer Signalling Network: a systems biology resource for integrative analysis of cancer data with Google Maps. *Oncogenesis* *4*, e160.

Laoui, D., Van Overmeire, E., Movahedi, K., Van den Bossche, J., Schouppe, E., Mommer, C., Nikolaou, A., Morias, Y., De Baetselier, P., and Van Ginderachter, J.A. (2011). Mononuclear phagocyte heterogeneity in cancer: Different

subsets and activation states reaching out at the tumor site. *Immunobiology* *216*, 1192–1202.

Mantovani, A., Marchesi, F., Malesci, A., Laghi, L., and Allavena, P. (2017). Tumour-associated macrophages as treatment targets in oncology. *Nat. Rev. Clin. Oncol.* *14*, 399–416.

Marichal, T., Tsai, M., and Galli, S.J. (2013). Mast cells: potential positive and negative roles in tumor biology. *Cancer Immunol. Res.* *1*, 269–279.

Martinez, F.O., Gordon, S., Locati, M., and Mantovani, A. (2006). Transcriptional profiling of the human monocyte-to-macrophage differentiation and polarization: new molecules and patterns of gene expression. *J. Immunol.* *177*, 7303–7311.

Marvel, D., and Gabrilovich, D.I. (2015). Myeloid-derived suppressor cells in the tumor microenvironment: expect the unexpected. *J. Clin. Invest.* *125*, 3356–3364.

Mittal, D., Gubin, M.M., Schreiber, R.D., and Smyth, M.J. (2014). New insights into cancer immunoediting and its three component phases--elimination, equilibrium and escape. *Curr. Opin. Immunol.* *27*, 16–25.

Moynihan, K.D., and Irvine, D.J. (2017). Roles for Innate Immunity in Combination Immunotherapies. *Cancer Res.* *77*, 5215–5221.

Murray, P.J., and Wynn, T.A. (2011). Obstacles and opportunities for understanding macrophage polarization. *J. Leukoc. Biol.* *89*, 557–563.

Le Novère, N., Hucka, M., Mi, H., Moodie, S., Schreiber, F., Sorokin, A., Demir, E., Wegner, K., Aladjem, M.I., Wimalaratne, S.M., et al. (2009). The Systems Biology Graphical Notation. *Nat. Biotechnol.* *27*, 735–741.

O’Hara, L., Livigni, A., Theo, T., Boyer, B., Angus, T., Wright, D., Chen, S.-H., Raza, S., Barnett, M.W., Digard, P., et al. (2016). Modelling the Structure and Dynamics of Biological Pathways. *PLoS Biol.* *14*, e1002530.

O’Sullivan, T., Saddawi-Konefka, R., Vermi, W., Koebel, C.M., Arthur, C., White, J.M., Uppaluri, R., Andrews, D.M., Ngiow, S.F., Teng, M.W.L., et al. (2012). Cancer immunoediting by the innate immune system in the absence of adaptive immunity. *J. Exp. Med.* *209*, 1869–1882.

Ostrand-Rosenberg, S., and Sinha, P. (2009). Myeloid-derived suppressor cells: linking inflammation and cancer. *J. Immunol.* *182*, 4499–4506.

Van Overmeire, E., Laoui, D., Keirsse, J., Van Ginderachter, J.A., and Sarukhan, A. (2014). Mechanisms Driving Macrophage Diversity and Specialization in Distinct Tumor Microenvironments and Parallelisms with Other Tissues. *Front. Immunol.* *5*, 127.

Palucka, K., and Banchereau, J. (2012). Cancer immunotherapy via dendritic cells. *Nat. Rev. Cancer* *12*, 265–277.

R Core Team (2013). R: A Language and Environment for Statistical Computing. R Found. Stat. Comput. Vienna, Austria *0*, {ISBN} 3-900051-07-0.

- Raza, S., Robertson, K.A., Lacaze, P.A., Page, D., Enright, A.J., Ghazal, P., and Freeman, T.C. (2008). A logic-based diagram of signalling pathways central to macrophage activation. *BMC Syst. Biol.* 2, 36.
- Theoharides, T.C., and Conti, P. (2004). Mast cells: the Jekyll and Hyde of tumor growth. *Trends Immunol.* 25, 235–241.
- Tirosh, I., Izar, B., Prakadan, S.M., Wadsworth, M.H., Treacy, D., Trombetta, J.J., Rotem, A., Rodman, C., Lian, C., Murphy, G., et al. (2016). Dissecting the multicellular ecosystem of metastatic melanoma by single-cell RNA-seq. *Science* (80-). 352, 189–196.
- Tokunaga, R., Zhang, W., Naseem, M., Puccini, A., Berger, M.D., Soni, S., McSkane, M., Baba, H., and Lenz, H.-J. (2018). CXCL9, CXCL10, CXCL11/CXCR3 axis for immune activation – A target for novel cancer therapy. *Cancer Treat. Rev.* 63, 40–47.
- Topalian, S.L., Drake, C.G., and Pardoll, D.M. (2015). Immune Checkpoint Blockade: A Common Denominator Approach to Cancer Therapy. *Cancer Cell* 27, 450–461.
- Torphy, R., Schulick, R., and Zhu, Y. (2017). Newly Emerging Immune Checkpoints: Promises for Future Cancer Therapy. *Int. J. Mol. Sci.* 18, 2642.
- Vesely, M.D., Kershaw, M.H., Schreiber, R.D., and Smyth, M.J. (2011). Natural innate and adaptive immunity to cancer. *Annu. Rev. Immunol.* 29, 235–271.
- Vivier, E., Ugolini, S., Blaise, D., Chabannon, C., and Brossay, L. (2012). Targeting natural killer cells and natural killer T cells in cancer. *Nat. Rev. Immunol.* 12, 239–252.
- Vo, M.-C., Nguyen-Pham, T.-N., Lee, H.-J., Jaya Lakshmi, T., Yang, S., Jung, S.-H., Kim, H.-J., and Lee, J.-J. (2017). Combination therapy with dendritic cells and lenalidomide is an effective approach to enhance antitumor immunity in a mouse colon cancer model. *Oncotarget* 8, 27252–27262.
- Wickham, H. (2009). *ggplot2 Elegant Graphics for Data Analysis*.

FIGURE AND TABLE LEGENDS

Figure 1. Map construction workflow and map structure. The scheme demonstrates the steps of meta-map construction starting from collection of cancer specific and innate-immune specific information about individual molecular interactions from scientific publications and databases, manual annotation and curation of this information (steps 1-4), then organisation of this formalized knowledge in form of cell-type specific maps (step 5), and finally integration the cell-type specific networks in one global meta-map of innate immune response in cancer with areas corresponding to biological processes, modules, pro- and anti-tumor polarisation. (step 6)

Figure 2. Cell type-specific maps. Cell type specific networks are visualized at the top-level view, the colourful background indicates boundaries of functional modules of the maps.

Figure 3. Structure of meta-map of innate immune response in cancer. **(A)**. Top view layout of the innate immune meta-map. Functional modules represent key processes involved in pro-tumor and anti-tumor activity of innate immunity in cancer, showed at different zoom levels (0 – polarization and biological processes, 1-functional modules, 2- signalling pathways, molecules, interaction types and annotation details. **(B)**. Signalling pathways in the meta-map structure in a browser window. **(C)**. The network of modules demonstrating cross-talks between biological processes represented on the meta-map. Nodes represent biological processes with the size associated to number of molecules in a process, color of the node is related to pro/anti- tumor polarization (see legend), interactions reflect cross-talk between the biological processes, the thickness of the edge is related to number of interactions and the color to the nature of interactions.

Figure 4. Visualization of modules activity scores demonstrates functional difference between Pro- and Anti-tumor polarized macrophages cells in the context of maps. **(A)**. Macrophages single cells in PC1 and PC2 coordinates space. Two groups, the first and the fourth quartile of distribution along the IC1 axis, are colored distinctly in blue and black. Staining of the Macrophage cell type-specific map with modules activity scores calculated from single cell RNAseq expression data for **(B)**. Macrophages Groups 1 (Anti-tumor) and **(C)**. Macrophages Groups 2 (Pro-tumor) cells. Staining of the innate immune meta-map with modules activity scores calculated from single cell RNAseq expression data for **(D)**. Macrophages Groups 1(Anti-tumor) and **(E)**. Macrophages Groups 2 (Pro-tumor) cells. Color code: Red–upregulated, green- downregulated module activity.

Figure 5. Visualization of modules activity scores using expression data from melanoma natural killers (NK) shows the possible pathways regulated tumor-killing abilities of NK cells in melanoma. NK single cells in PC1 and PC2 coordinates space. Two groups are colored distinctly in blue and black. **(A)**. Map staining of the NK cell type-specific map with modules activity scores calculated from single cell RNAseq expression data for **(B)**. NK Group 1. **(C)**. Heatmap of activity scores in signalling pathways of NK groups. Map staining of the innate immune response meta-map with modules activity scores for **(D)**. NK Group 1 (“tumor killing”) with a zoom into three signalling pathways relating the two upregulated modules: “Danger signal pathways” and “Exocytosis and phagocytosis” with main molecular players named . Color code: Red–upregulated, green- downregulated module activity.

Table 1. Hierarchical modular structure of innate immune response meta-map.

Table 2. Distribution of genes with positive ($z<0$) and negative ($z>0$) correlation with patient survival across functional meta-modules in innate immune response meta-map. Values indicate number of genes.

Table 3. Comparison of innate immune response representation in different pathway databases

STAR METHODS

Maps access and data availability

The cell type-specific maps and meta-map of innate immune response in cancer are freely available for downloading from the NaviCell web page (<http://navicell.curie.fr/pages/maps>) in several exchange formats. The composition of map signaling pathways, modules and meta-modules is provided in a form of GMT files (Supplemental Tables 2 and 3

respectively) suitable for further functional data analysis.

Map and model

Cell type-specific maps and meta-map of innate immune response in cancer were created using the methodology developed in (Kondratova et al., 2016)(Kuperstein et al., 2015). The maps are drawn in CellDesigner diagram editor (Kitano et al., 2005) using Process Description (PD) dialect of Systems Biology Graphical Notation (SBGN) syntax which is based on Systems Biology Markup Language (SBML) (Le Novère et al., 2009). The data model, includes the following molecular objects: proteins, genes, RNAs, antisense RNAs, simple molecules, ions, drugs, phenotypes, complexes. These objects can play role of reactants, products and regulators in a connected reaction network. Edges on the maps represent biochemical reactions or reaction regulations of various types. Different reaction types represent posttranslational modifications, translation, transcription, complex formation or dissociation, transport, degradation and so on. Reaction regulations include catalysis, inhibition, modulation, trigger and physical stimulation. The naming system of the maps is based on HUGO identifiers for genes, proteins, RNAs and antisense RNAs and CAS identifiers for drugs, small molecules and ions.

Manual literature mining

Molecular interactions reported in the scientific articles were manually curated and the information extracted from the papers was used for reconstruction and annotation the maps. Three types of articles were used for map annotation: (i) experimental innate-immunity specific articles directly or indirectly confirming molecular interactions based on mammalian experimental data; (ii) review articles; (iii) experimental articles from non-immune cells that helped to complement the mechanisms present in immune cells (3% of the literature used for the map). In addition, pathways databases were used to retrieve information of the canonical pathways reported for the innate immune signalling general pathway databases (e.g. KEGG, REACTOME, SPIKE SignaLink, EndoNET) or in the immune system-specialized resources such as VirtuallyImmune (<http://www.virtuallyimmune.org>) and InnateDB (www.innatedb.com).

Map tagging system

The annotation of each molecular object on the maps (protein, gene, RNA, small molecule etc) includes several tags indicating participation of the object in: signalling pathways (tag CASCADE:NAME), functional modules (tag MODULE:NAME) and cell type-specific map (tag: MAP:NAME). Each CASCADE obtains the name of the initiating ligand or receptor, in case when several ligands are acting through the same receptor (Supplemental Figure 4). The tags are converted into the links by the NaviCell factory in the process of online map version generation. The links allow to trace participation of entities in different cell type-specific maps and the sub-structure of the same map (pathway, module, biological process) and also facilitate shuttling between these structures.

Map entity annotation

The annotation panel followed the NaviCell annotation format of each entity of the maps includes sections ‘Identifiers’, ‘Maps_Modules’, ‘References’ and ‘Confidence’ as detailed in (Kuperstein et al., 2015). ‘Identifiers’ section provides standard identifiers and links to the corresponding entity descriptions in HGNC, UniProt, Entrez, SBO, GeneCards and

cross-references in REACTOME, KEGG, Wiki Pathways and other databases. ‘Maps_Modules’ section includes tags of modules, meta-modules, and cell type-specific maps in which the entity is implicated (see above). ‘References’ section contains links to related publications. Each entity annotation is represented as a post with extended information on the entity.

Generation of NaviCell map with NaviCell factory

CellDesigner map annotated in the NaviCell format is converted into the NaviCell web-based front-end, which is a set of html pages with integrated JavaScript code that can be launched in a web browser for online use. HUGO identifiers in the annotation form allow using NaviCell tool for visualization of omics data. Detailed guide of using the NaviCell factory embedded in the BiNoM Cytoscape plugin is provided at <https://navicell.curie.fr/doc/NaviCellMapperAdminGuide.pdf>.

High-throughput data source and software

Normalized melanoma dataset from GEO (GSE72056)(Tirosh et al., 2016) were transformed into log expression levels and mean centred. The exploratory analysis and statistical testing was performed and visualized using R packages (ggplot2, stats, pheatmap) (Kolde, 2012; R Core Team, 2013; Wickham, 2009) then MATLAB ICA implementation of FastICA algorithm (Hyvärinen and Oja, 2000) and icasso package (Himberg and Hyvärinen, 2003) to improve the stability. Colored map images were obtained using function “Stain CellDesigner map” from BiNoM Cytoscape plugin (Bonnet et al., 2013) using .xml map files and the mean expression from the analysis described below.

Analytical pipeline

Group definitions

Independent components analysis, which computes numerical vectors of weights that represent independent factors maximizing non-Gaussian signal, was used to sort the data points along the axis of the first independent component (as it was the only stable dimension according to icasso stability analysis). We divided the NK single cells in half depending on the first independent component (IC1) projection score such that Group 1 had positive projection scores and the Group 2 has negative projection scores. As far as Macrophage single cells are concerned, the distribution of the data was remarkably bimodal along the IC1. In order to best interpret the “extreme” tendencies of the cells placed on the opposite side of IC we selected the first and the last quartile of the macrophage scores of IC1 projection. The distinction of the groups plotted in first and the second principal components space (PC1 and PC2) can be seen in Supplemental Figures 4A and 5A.

Activity scores

For groups defined as described above, following procedure was applied for both NK and Macrophages. For each module, 50% of most variant genes were retained without distinction of cells into groups. Subsequently, cells were selected depending on their group attribution and mean of genes were computed per module per group in each map for visualization purposes (map staining and heatmaps).

In order to assess statistically the possible differences between groups, we compared genes retained for each module

between groups of cells with a t-test. The p-values of the t-test were reported on heatmaps with standard code of significance ($*** < 0.001$, $** < 0.01$, $* < 0.05$, $. < 0.1$). The same mean values per group and per module were plotted on the maps.

Enrichment of innate immune response meta-map with patient survival-correlating genes

The data on pan-cancer meta-analysis of expression signatures from ~18,000 human tumors with overall survival outcomes across 39 malignancies were used (Gentles et al., 2015). The 6323 genes with the z-scores (p-value < 0.05) indicating correlation to patient survival were retrieved (Gentles et al., 2015) and overplayed with the gene list from the innate immune response meta-map. Enrichment of the meta-map with the genes significantly positively or negatively-correlated with patient survival were detected using the Chi-square test, using p-value threshold < 0.001 .

SUPPLEMENTAL INFORMATION

Supplemental Figure 1. Entity annotation structure page in NaviCell format.

Supplemental Figure 2. Comparison of InnateDB, REACTOME, KEGG and Innate immune meta-map databases based on gene names content. (A) Visualization of distribution of 188 unique genes from the innate immune response meta-map across map modules. The content of the map was compared with innate immune-related sub-set of pathways from Innate DB, KEGG and REACTOME and 188 unique genes were identified and visualized. (B) Enrichment of functional modules on innate immune response meta-map with the 188 unique genes (percentage) The p-value of the Chi-square-test is reported following the code: $*** < 0.001$, $** < 0.01$, $* < 0.05$.

Supplemental Figure 3. Comparison of InnateDB, REACTOME, KEGG and Innate immune meta-map databases based on publications used for map annotation. (A) Venn diagram showing intersection of the publications annotating the selected pathways (see the main text) from the four different databases. Distribution of (B) publication years and (C) different types of journals annotating the selected pathways (see the main text) from InnateDB, REACTOME and the Meta-map. The peak in the graph indicates papers from 2010-2015 years of publication is indicated by arrow. (D) Relative use of different types of journals for annotation of the selected pathways (see the main text) from InnateDB, REACTOME and the Meta-map databases.

Supplemental Figure 4. Sup-populations study and calculation of modules activity scores using expression data from melanoma macrophage cells. Activity scores of Macrophages in the two groups for (A) cell type-specific map and for (B) meta-map. The p-value of the t-test between gene expression is reported following the code: $*** < 0.001$, $** < 0.01$, $* < 0.05$, $. < 0.1$

Supplemental Figure 5. Sup-populations study and calculation of modules activity scores using expression data from melanoma natural killers (NK) cells. Heatmap of activity scores of each group in modules of (A) cell-type-specific map and (B) meta-map. The p-value of the t-test between gene expression is reported following the code: $*** < 0.001$, $** < 0.01$, $* < 0.05$, $. < 0.1$

Supplemental Figure 6. Meta-map as a potential source of prognostic signatures for patient survival. (A)

Visualization of mean z-scores of meta-modules. Blue zones are enriched by genes with a positive correlation to patient survival, yellow zones are enriched by genes correlated with negative patient survival.

Supplemental Table 1. Modular structure of cell type-specific maps.

Supplemental Table 2. List and content of signalling pathways on innate immune meta-map. PROVIDED AS A SEPARATE FILE

Supplemental Table 3. List and content of modules and meta-modules on innate immune meta-map. PROVIDED AS A SEPARATE FILE

Supplemental Table 4. List of pathways and genes content from Innate DB, KEGG and REACTOME used for comparison of innate immune response with these resources. FULL TABLE PROVIDED AS A SEPARATE FILE

Supplemental Table 5. List of unique genes from innate immune response meta-map comparing to gene lists in pathways selected from InnateDB, KEGG and REACTOME for comparison. PROVIDED AS A SEPARATE FILE

Supplemental Table 6. List genes from innate immune response meta-map positively or negatively-correlated with patient survival. PROVIDED AS A SEPARATE FILE

FIGURES & TABLES

1 RETRIEVING CANCER AND INNATE IMMUNE CELLS RELATED PUBLICATIONS



Figure 1

2 CLASSIFYING INFORMATION INTO CELL-SPECIFIC GROUPS



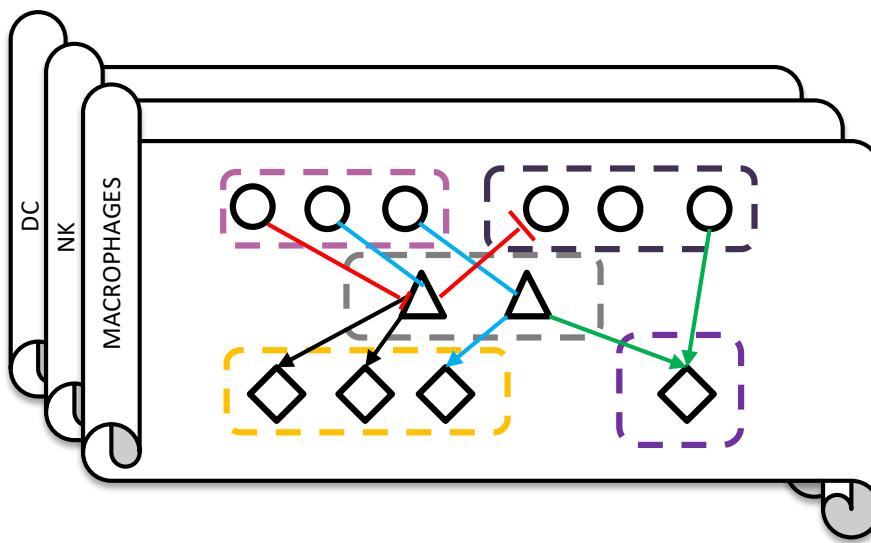
3 MANUAL CURATION OF MOLECULAR INTERACTIONS



4 QUALITY AND REPRODUCIBILITY CHECK



5 KNOWLEDGE ORGANISATION INTO CELL SPECIFIC NETWORK MAPS



ORGANIZATION OF LAYERS

- Molecular type (horizontal)**
- inducers
 - △ intermediates
 - ◇ effectors

- Pathways (up to bottom)**
- activation
 - ↔ inhibition
 - molecular flow

- Functional modules (area)**
- functional module

6 INTEGRATION INTO A META MAP

INHERITS FEATURES OF CELL SPECIFIC MAPS AND IN ADDITION CONTAINS:

- PRO- / ANTI-TUMOR ZONES
- ADDITIONAL NEUTROPHIL AND MAST CELL INTRACELLULAR INTERACTIONS
- CELL-CELL INTERACTIONS
- BIOLOGICAL PROCESSES (GROUPS OF FUNCTIONAL MODULES)

META MAP OF INNATE IMMUNE RESPONSE IN CANCER

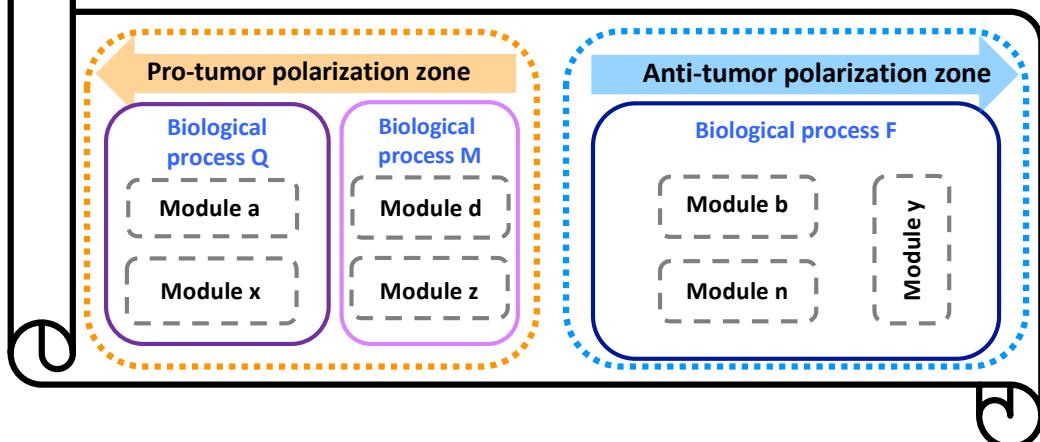


Figure 2

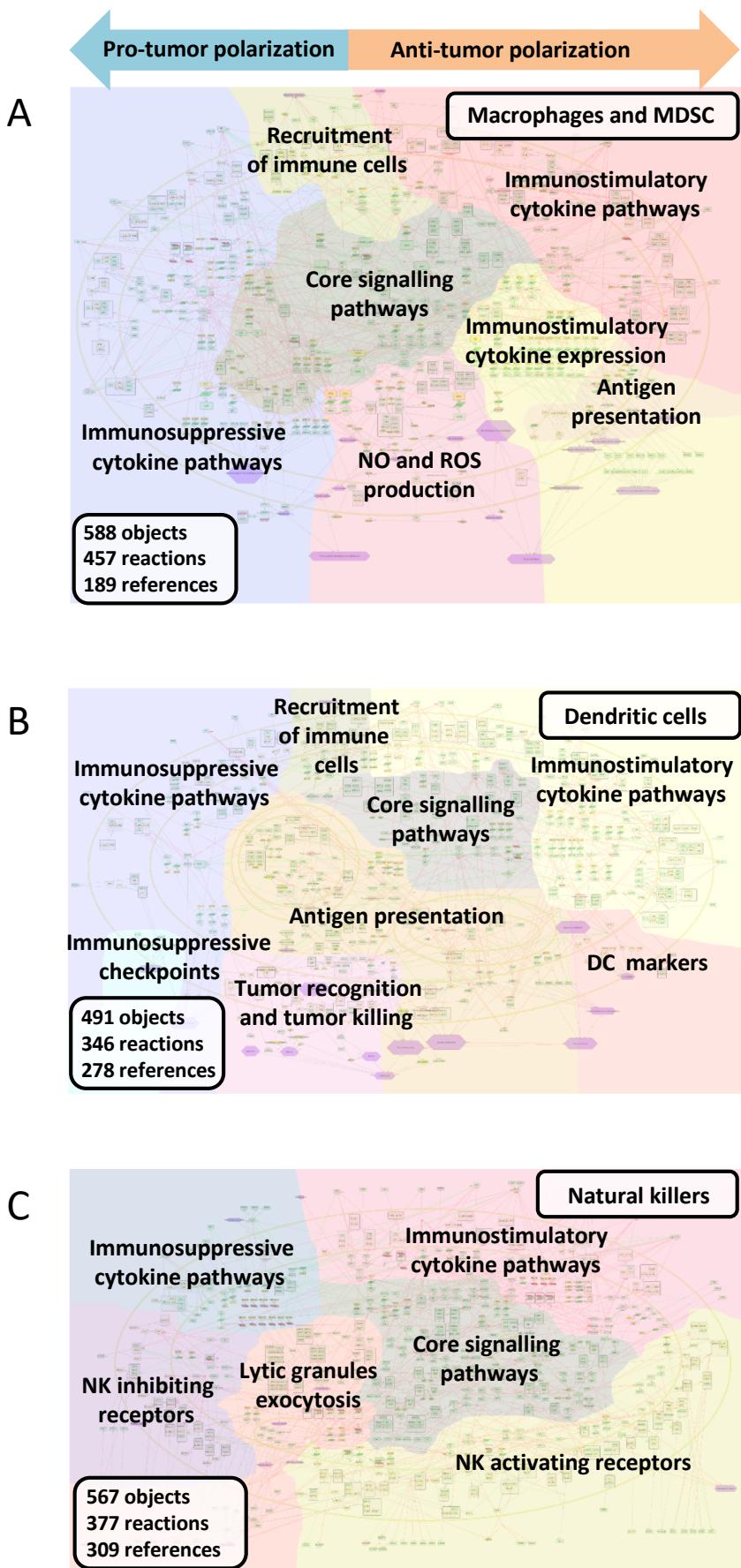
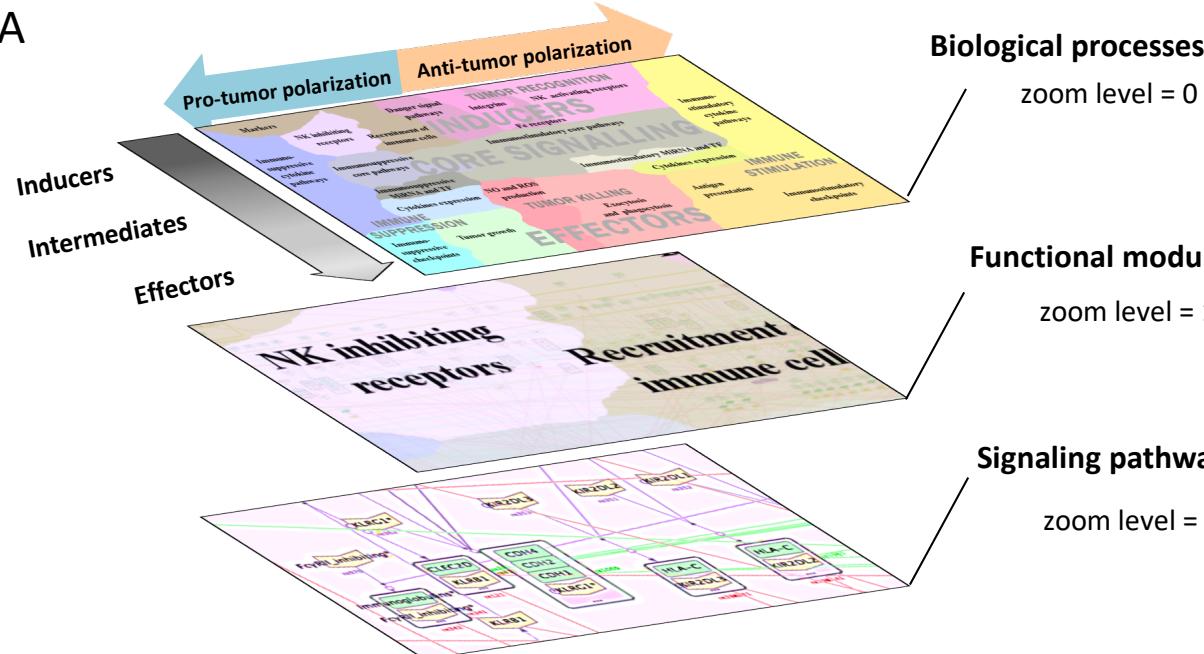
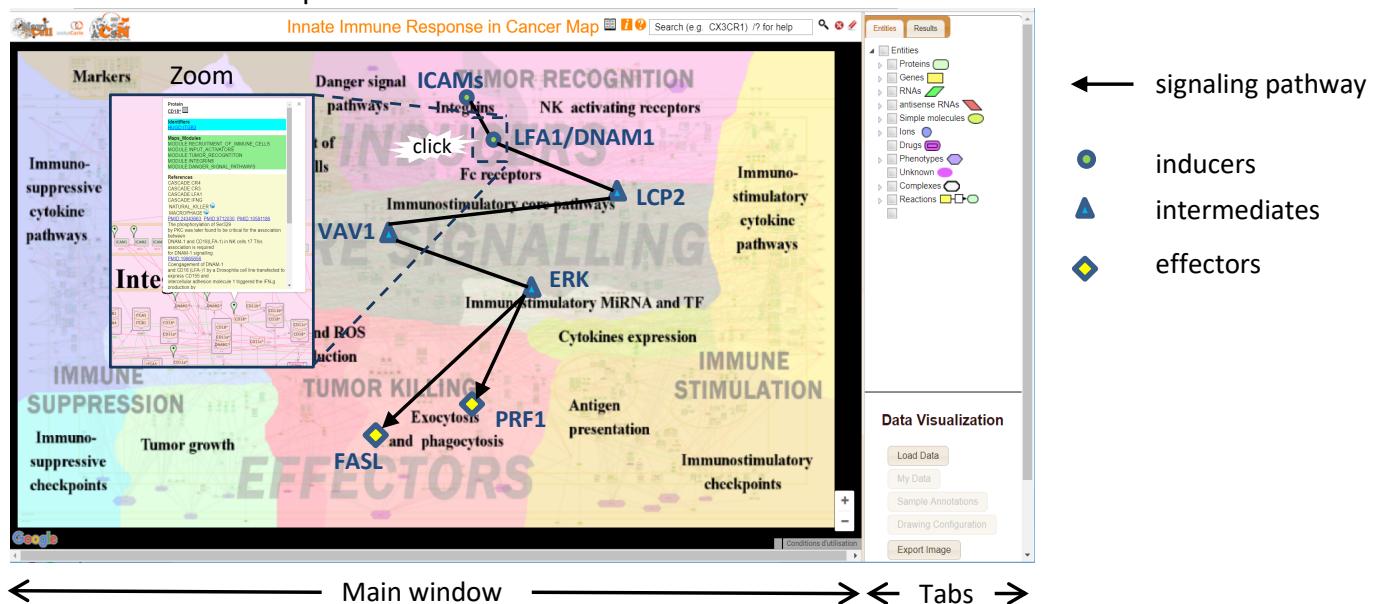


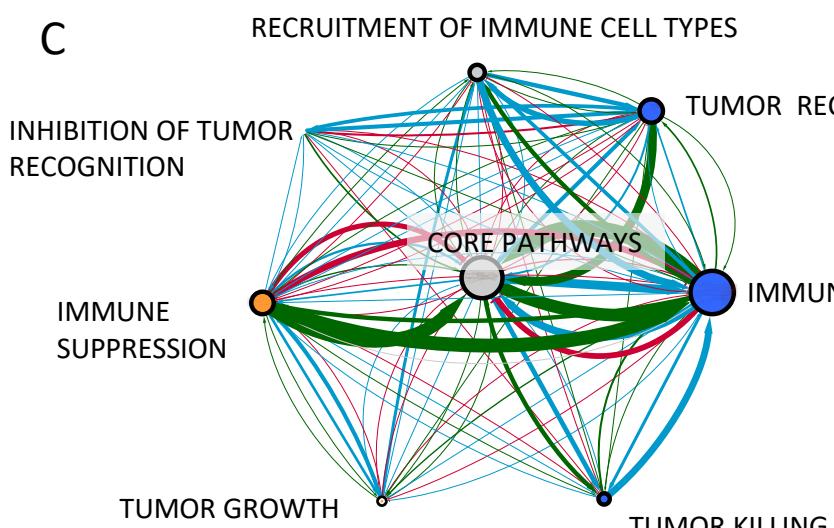
Figure 3



B Screenshot of the map in the browser



C



Type of biological process

- anti-tumor (blue)
- neutral (grey)
- pro-tumor (orange)

Type of interaction

- activation (green arrow)
- inhibition (red arrow)
- molecular flow (blue line)

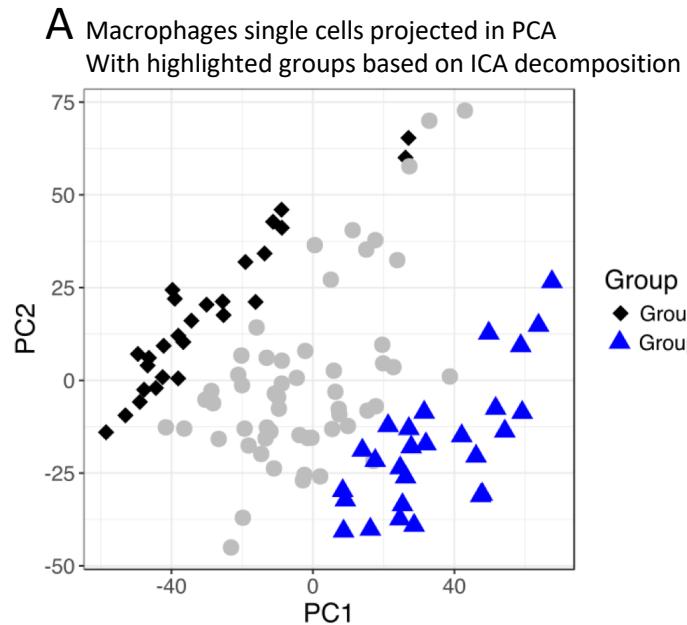
Number of molecules

- 80 (small circle)
- 652 (large circle)

Number of interactions

- 1 (thin black bar)
- 101 (thick black bar)

Figure 4



SUMMARY:

Upregulated modules in **Anti-tumor Gr 1**:

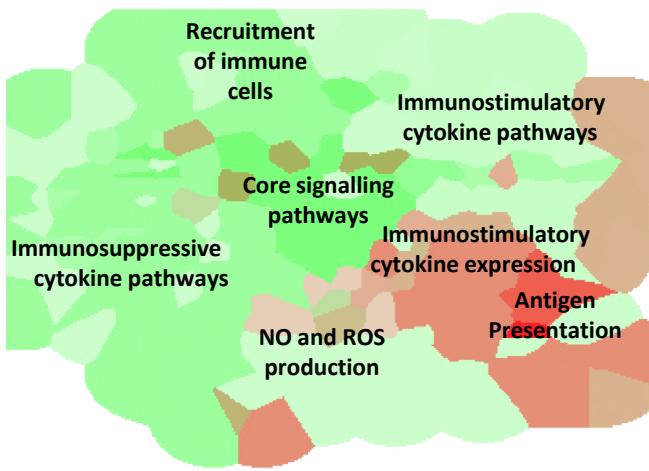
- Antigen presentation (cell-specific and meta map)
- Immunosuppressive checkpoints
- Danger signal modules
- Immunostimulatory MiRNA and TF

Upregulated modules in **Pro-tumor Gr 2**:

- Tumor Growth
- Immunosuppressive cytokine expression (cell-specific and meta map)
- Recruitment of immune cells module
- Core signalling pathways (cell-specific map)

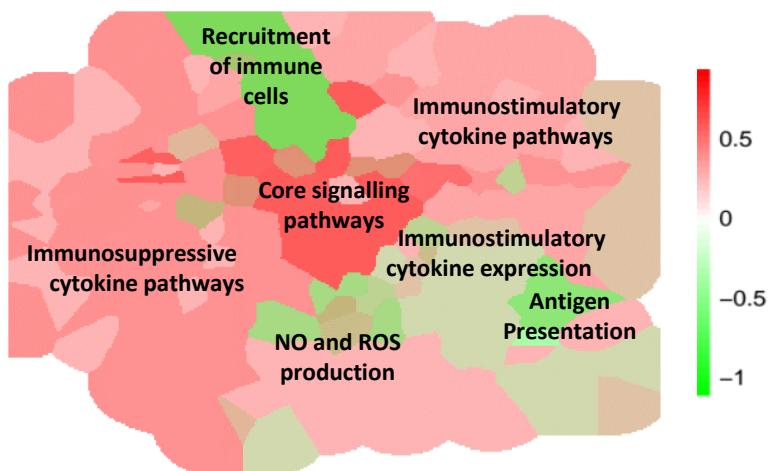
B

Macrophages cell type-specific map: Anti-tumor Group 1



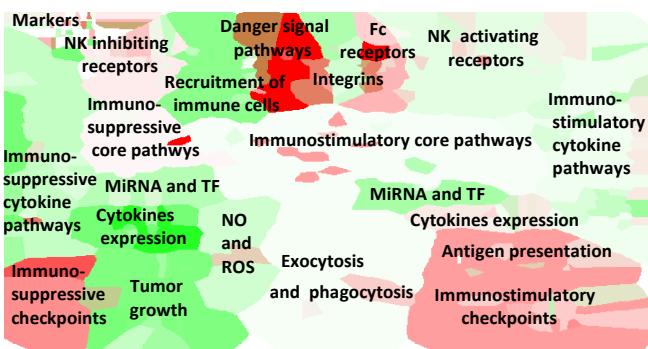
C

Macrophages cell type-specific map: Pro-tumor Group 2



C

Innate immune response meta-map: Anti-tumor Group 1



D

Innate immune response meta-map: Pro-tumor Group 2

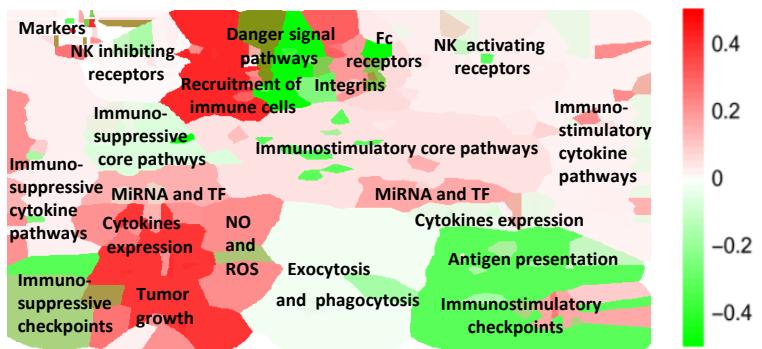


Figure 5

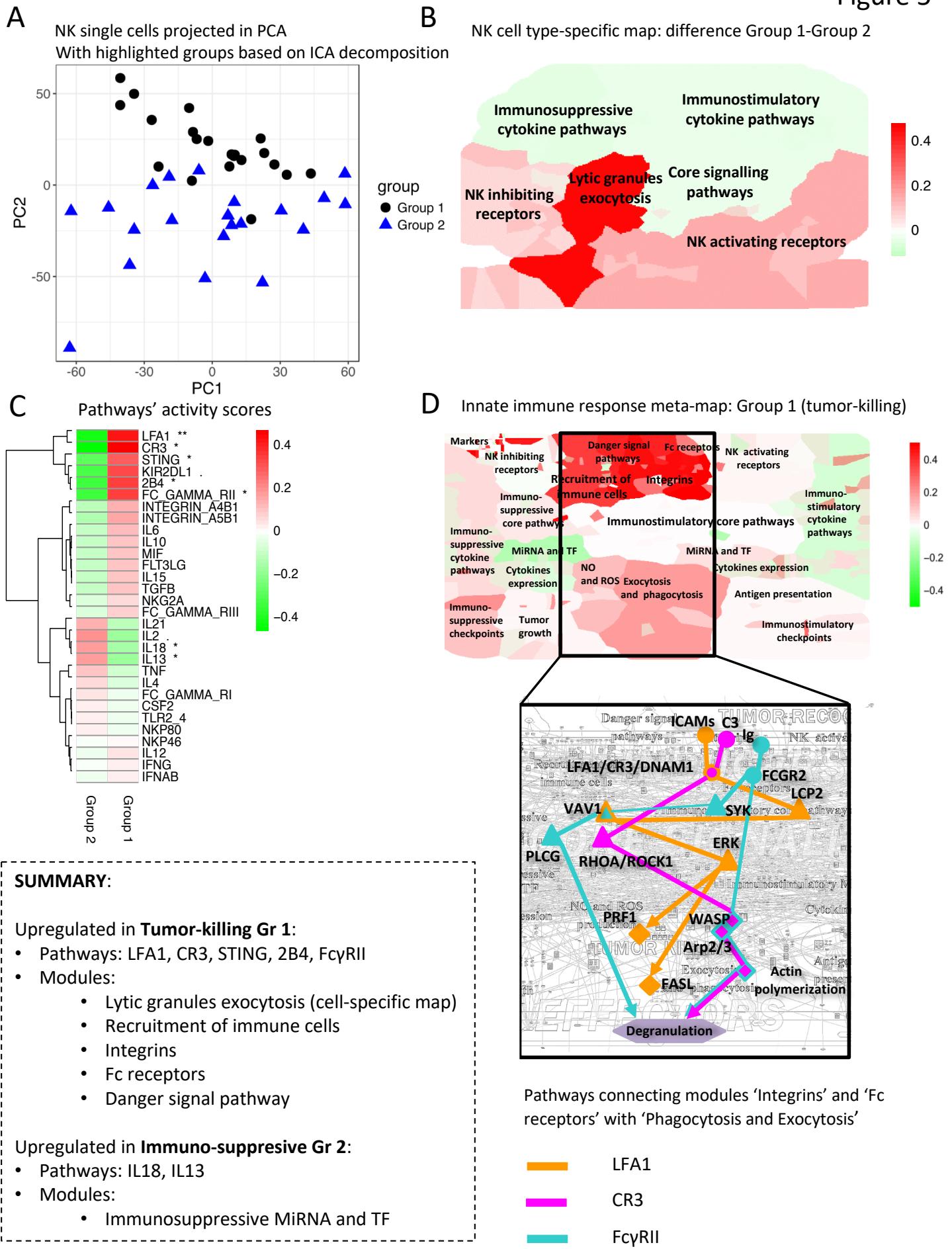


Table 1: Structure and content of Innate immune meta-map

Zones/Metamodule/Module	Chemical Species (Entities)	Proteins	Genes	RNAs	asRNAs	Reactions	References
Zone: Pro-tumor polarization							
INHIBITION OF TUMOR RECOGNITION							
NK INHIBITING RECEPTORS							
IMMUNE SUPPRESSION	35	23	1	1	0	14	57
IMMUNOSUPPRESSIVE CYTOKINE PATHWAYS	109	46	10	11	3	67	114
IMMUNOSUPPRESSIVE CYTOKINE EXPRESSION	55	19	14	14	0	36	75
IMMUNOSUPPRESSIVE CHECKPOINTS	8	7	0	0	0	8	13
CORE SIGNALLING PATHWAYS							
IMMUNOSUPPRESSIVE CORE PATHWAYS	43	23	5	5	1	25	54
MIRNA TF IMMUNOSUPPRESSIVE	77	20	23	14	12	48	62
TUMOR GROWTH	60	42	8	8	0	71	58
TUMOR GROWTH							
Zone: Anti-tumor polarization							
TUMOR RECOGNITION							
NK ACTIVATING RECEPTORS							
DANGER SIGNAL PATHWAYS	114	45	16	14	6	72	115
FC RECEPTEORS	60	30	2	1	0	36	66
INTEGRINS	18	12	0	0	0	8	37
IMMUNE STIMULATION							
IMMUNOSTIMULATORY CYTOKINE PATHWAYS	152	74	18	18	3	92	193
IMMUNOSTIMULATORY CYTOKINE EXPRESSION	43	17	12	11	1	27	109
ANTIGEN PRESENTATION AND IMMUNOSTIMULATORY CHECKPOINTS	99	65	6	6	0	91	152
CORE SIGNALLING PATHWAYS							
IMMUNOSTIMULATORY CORE PATHWAYS	184	93	6	6	114		244
MIRNA TF IMMUNOSTIMULATORY	50	17	12	10	5	33	60
TUMOR KILLING							
LYtic GRANULES EXOCYTOSIS AND PHAGOCYTOSIS	73	39	6	6	5	50	75
NO ROS PRODUCTION	33	10	4	4	0	23	44
Cell type specific markers							
MARKERS							
MARKERS MACROPHAGE							
MARKERS NK	22	10	6	6	0	0	8
MARKERS MAST	10	10	0	0	0	0	36
MARKERS DC	6	6	0	0	0	0	9
MARKERS NEUTROPHILE	16	14	0	2	0	0	14
MARKERS MDSC	11	11	0	0	0	0	15
MARKERS	9	9	0	0	0	0	9
Recruitment							
RECRUITMENT OF IMMUNE CELLS							
RECRUITMENT OF IMMUNE CELLS							
103	48	17	17	0	93		83
META-MAP							
1466	582	162	152	20	1084		820

Table 2. Distribution of genes with positive ($z<0$) and negative ($z>0$) correlation with patient survival across functional meta-modules in innate immune response meta-map. Values indicate number of genes.

Innate immune map meta-module	Mean z-score	Positive correlation with patient survival	Negative correlation with patient survival
TUMOR GROWTH	1.3	12	26
INHIBITION OF TUMOR RECOGNITION	-1.86	18	6
TUMOR RECOGNITION	-1.56	67	28
RECRUITMENT OF IMMUNE CELLS	-0.94	29	14
IMMUNE STIMULATION	-0.53	122	87
TUMOR KILLING	-0.5	25	29
CORE SIGNALLING PATHWAYS	-0.46	114	84
IMMUNE SUPPRESSION	-0.33	39	24

Table 3: Comparison of innate immune response representation in different pathway databases

Feature/Database	KEGG	REACTOME	InnateDB	Innate immune response in cancer resource
Cancer specificity	-	-	-	+
Cell-type specificity	-	+/-	+/-	+
Cross-talks between pathways	+/-	+/-	+/-	+
Hierarchical organization of knowledge	-	+	-	+
Semantic zooming	-	+	-	+
Data visualization	-	+	-	+

Supplementary figures & tables

Figure S1

Protein IRF1

Identifiers

HUGO:IRF1

Maps_Modules

MODULE:MACROPHAGE
MODULE:NK
MODULE:CORE_SIGNALING
MODULE:MIRNA_TF_ACTIVATION
CASCADE:IL2
CASCADE:TNF
CASCADE:IFNAB
CASCADE:IFNG
CASCADE:TLR2_4

References

PMID:11399519

STAT1, IRF1 and NF- κ B interact with NOS2 promoter and cooperatively activate NOS2 expression in macrophages downstream of IFNG.

PMID:10820262

Probably IFNG induces SOCS1 expression via IRF1 upregulation downstream of STAT1

PMID:18345002

TNF induces IRF1 expression both through TNFR1 and TNFR2.

TNF induced IFNB expression through IRF1

PMID:12417340

IRF-8/ICSBP and IRF-1 cooperatively stimulate mouse IL-12 p40 promoter activity in macrophages.

IRF-1 can be acetylated by p300. p300 is recruited to the IL-12p40 promoter depending on both ICSBP and IRF-1 and acts as coactivator.

PMID:16597464

IRF-8 and IRF-1 are the target genes in activated macrophages.

The expression levels of CXCL16, H28, IL-17R, LIF, MAP4K4, MMP9, MYC, PCDH7, PML, and SOCS7 were significantly increased in macrophages extracted from WT mice following activation for 4 h with IFNG and LPS. However, no changes in the expression of these genes were observed in cells extracted from IRF-8,

IRF1@INNATE_IMMUNE_META_CELL

Modifications:

In compartment: INNATE_IMMUNE_META_CELL

1. IRF1@INNATE_IMMUNE_META_CELL
2. IRF1|ace@INNATE_IMMUNE_META_CELL

Participates in complexes:

Participates in reactions:

As Reactant or Product:

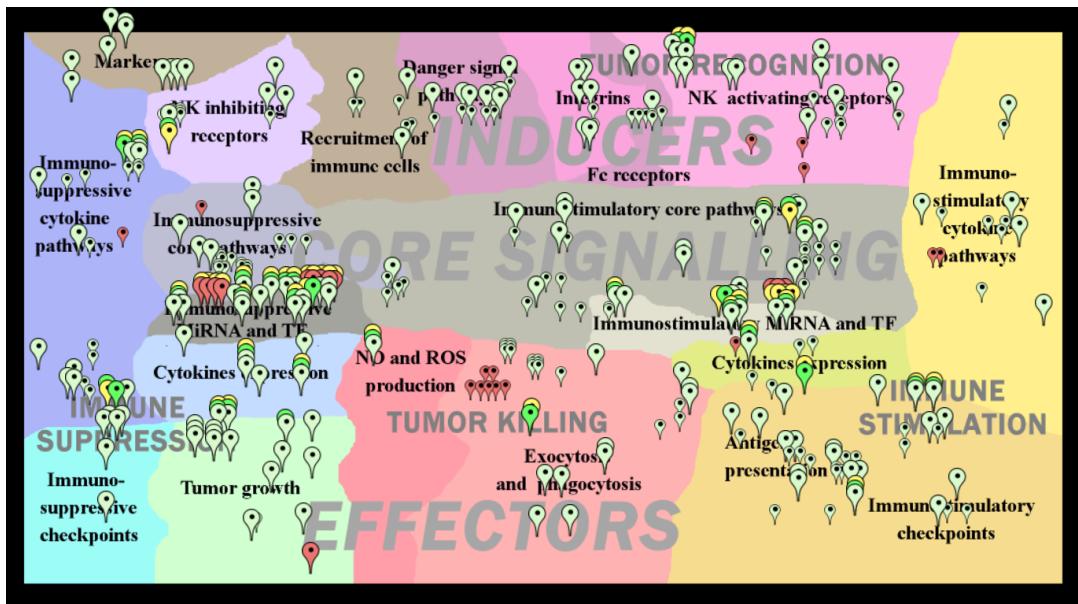
1. rIRF1@INNATE_IMMUNE_META_CELL → rIRF1@INNATE_IMMUNE_META_CELL
2. IRF1@INNATE_IMMUNE_META_CELL → rIRF1|ace@INNATE_IMMUNE_META_CELL

As Catalyst:

1. gIDO1@INNATE_IMMUNE_META_CELL → rIDO1@INNATE_IMMUNE_META_CELL
2. gCXCL16@INNATE_IMMUNE_META_CELL → rCXCL16@INNATE_IMMUNE_META_CELL
3. gNOS2@INNATE_IMMUNE_META_CELL → rNOS2@INNATE_IMMUNE_META_CELL
4. gIL12p40*@INNATE_IMMUNE_META_CELL → rIL12p40*@INNATE_IMMUNE_META_CELL
5. gIFNB*@INNATE_IMMUNE_META_CELL → rIFNB*@INNATE_IMMUNE_META_CELL
6. gMMP9@INNATE_IMMUNE_META_CELL → rMMP9@INNATE_IMMUNE_META_CELL
7. gCIITA@INNATE_IMMUNE_META_CELL → rCIITA@INNATE_IMMUNE_META_CELL
8. gSOCS1@INNATE_IMMUNE_META_CELL → rSOCS1@INNATE_IMMUNE_META_CELL
9. gMYC@INNATE_IMMUNE_META_CELL → rMYC@INNATE_IMMUNE_META_CELL
10. gTRAIL*@INNATE_IMMUNE_META_CELL → rTRAIL*@INNATE_IMMUNE_META_CELL

Figure S2

A



B

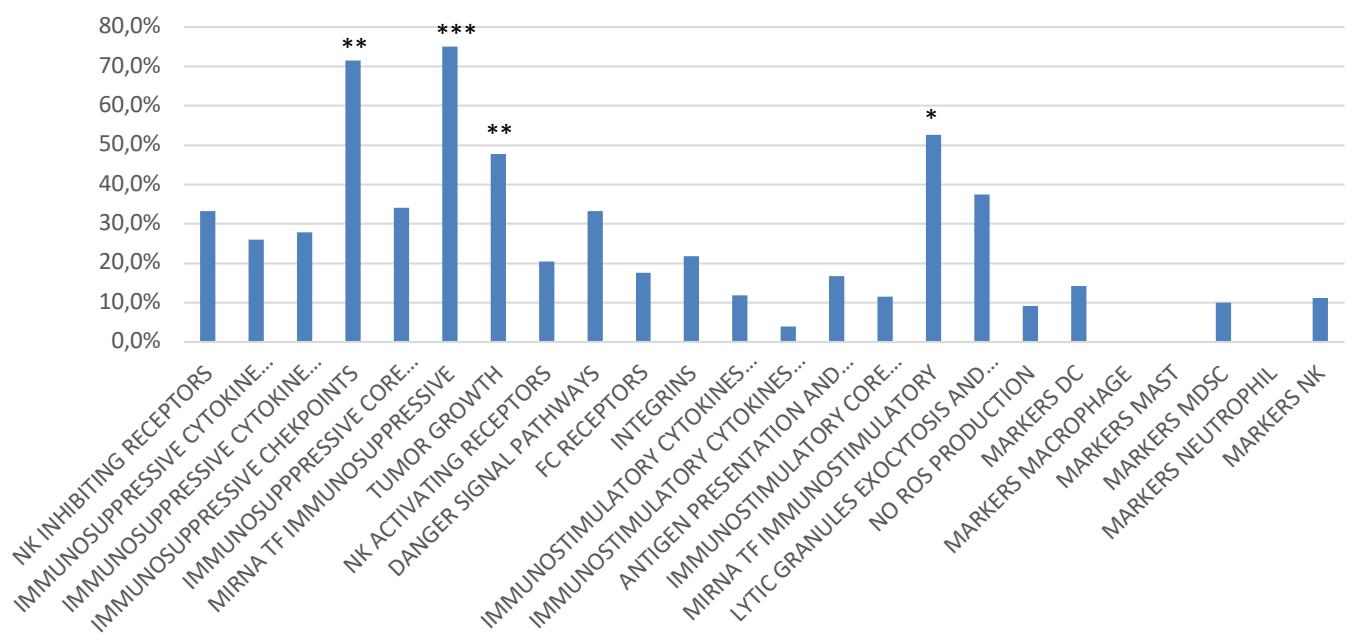
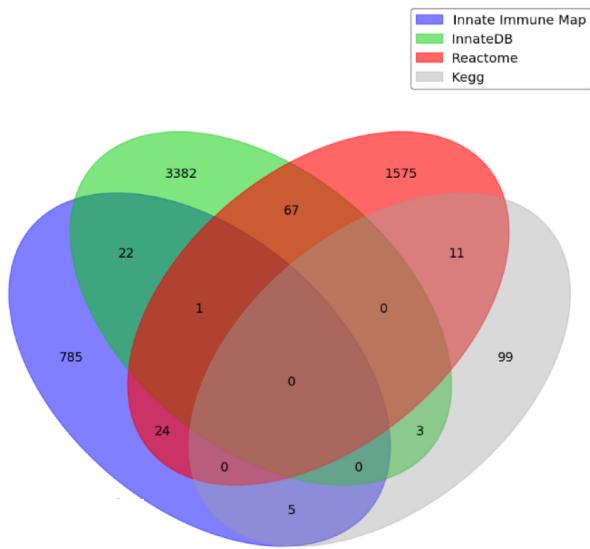
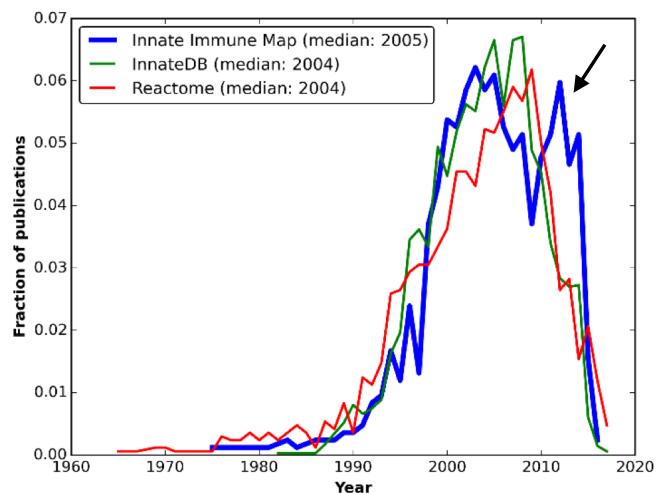


Figure S3

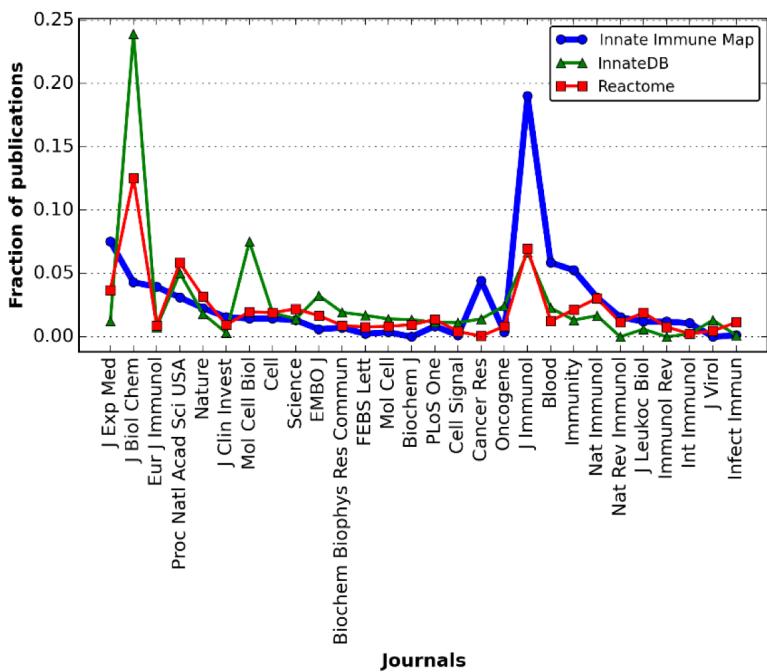
A



B



C



D

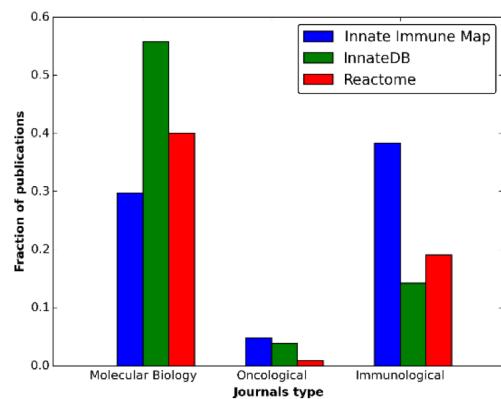


Figure S4

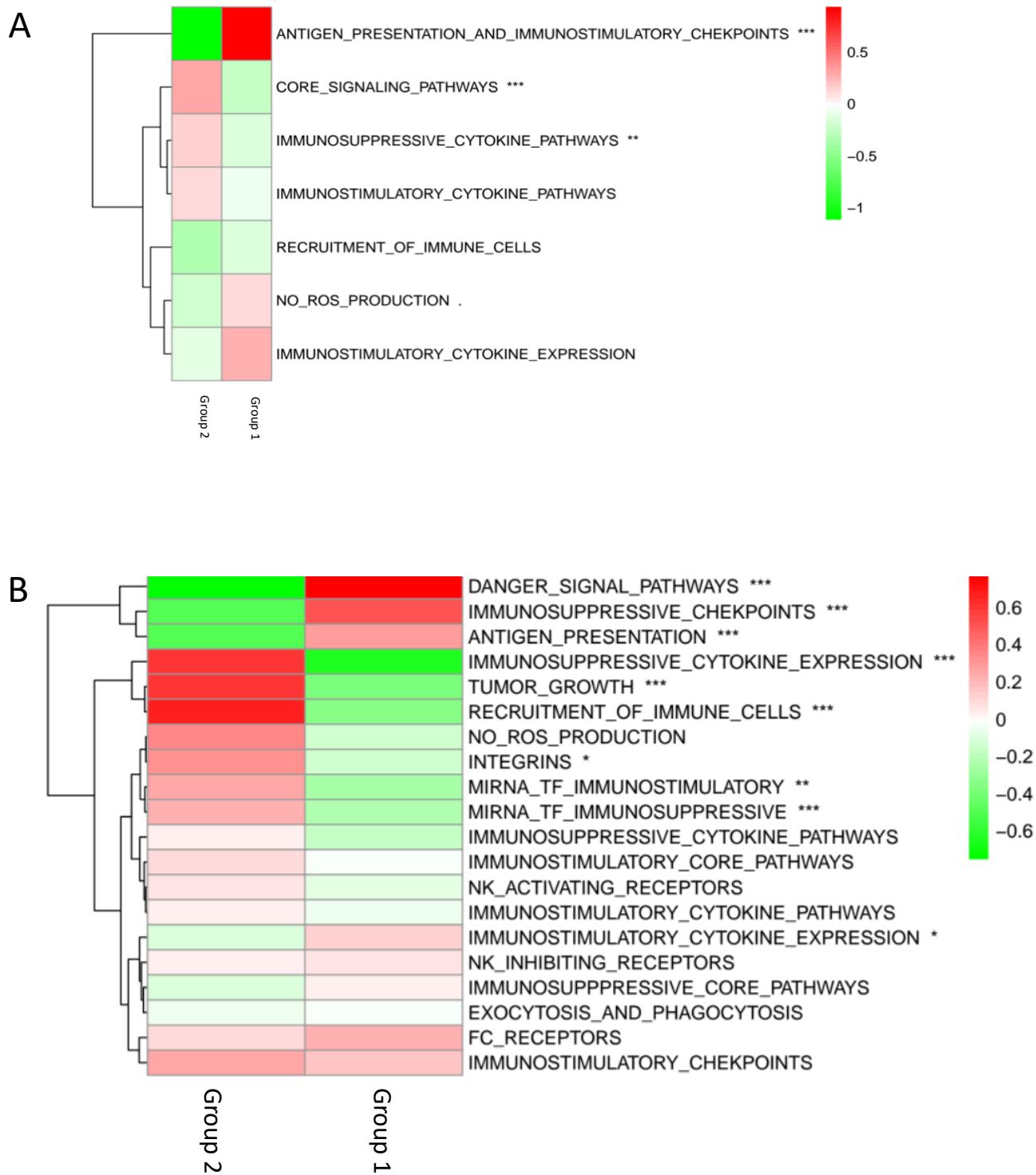


Figure S5

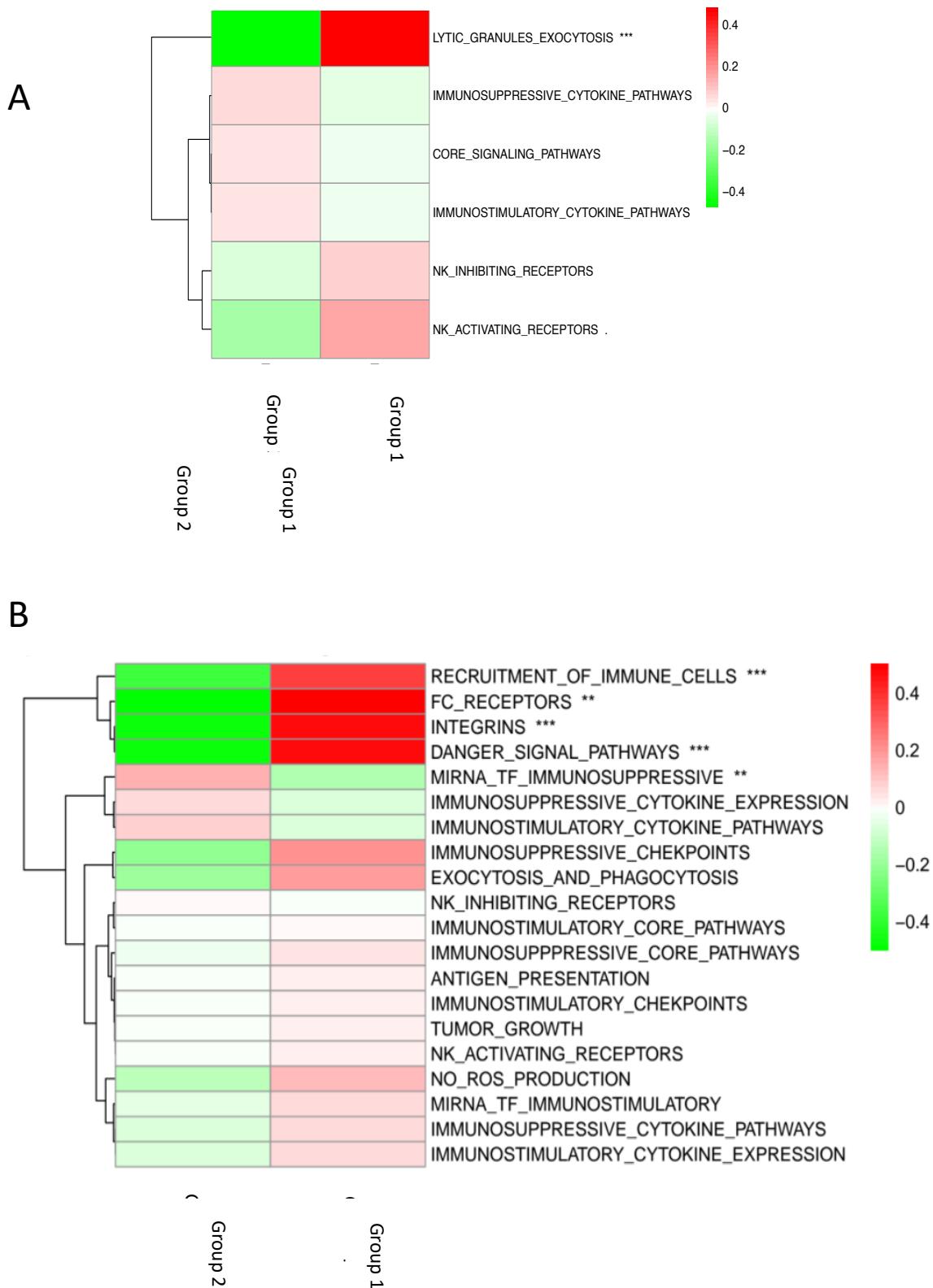
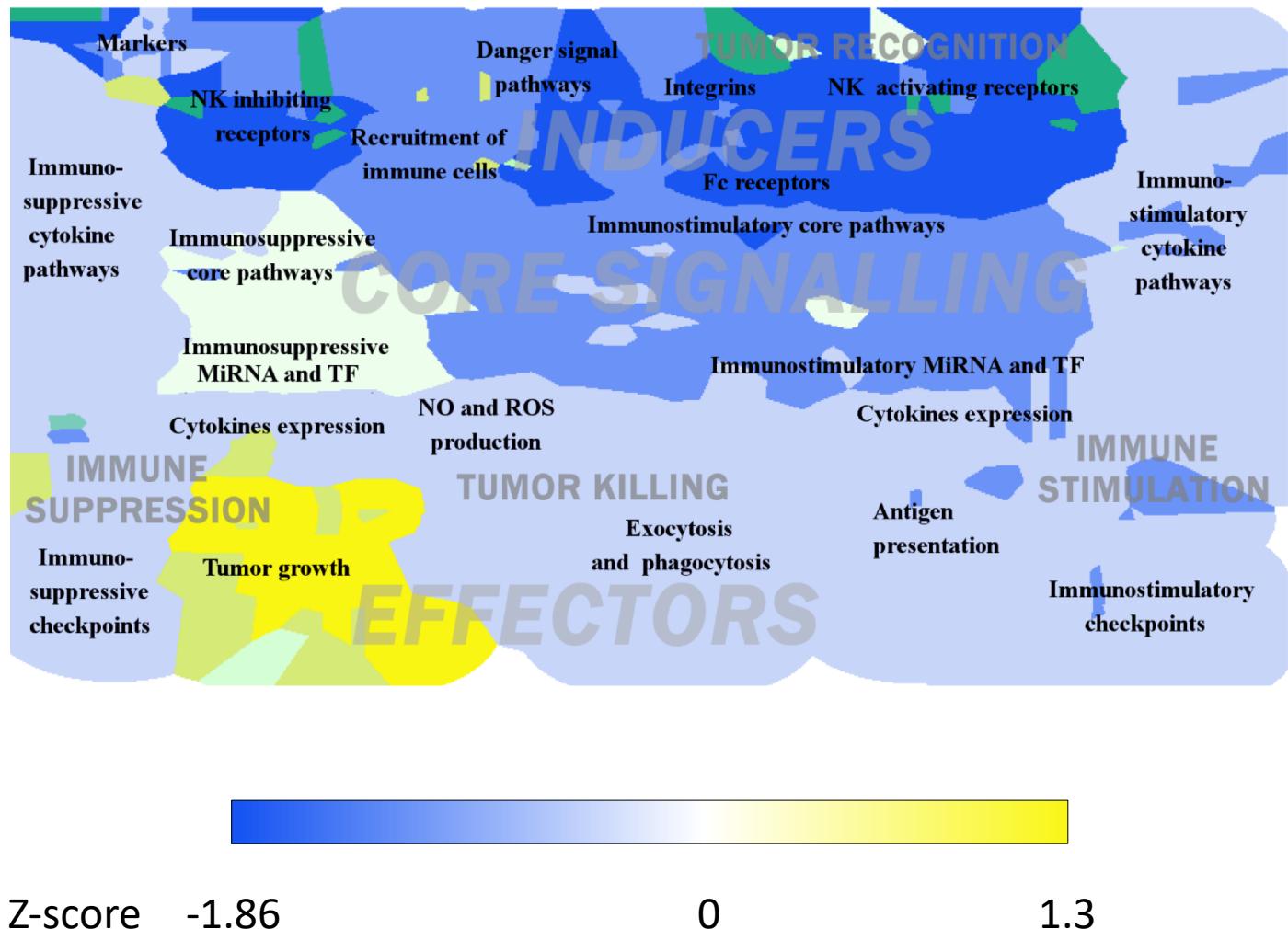


Figure S6



Supplemental Table 1: Structure and content of cell-type specific innate immune maps

Map/Module	Chemical Species	Proteins	Genes	RNAs	asRNAs	Reactions	References
Macrophages and MDSC	588	217	95	95	4	457	189
RECRUITMENT OF IMMUNE CELLS	37	15	6	6	0	29	18
NO ROS PRODUCTION	54	20	6	6	0	37	25
IMMUNOSTIMULATORY CYTOKINE PATHWAYS	92	55	10	9	1	50	75
IMMUNOSTIMULATORY CYTOKINE EXPRESSION	81	31	20	21	0	76	35
ANTIGEN PRESENTATION AND IMMUNOSTIMULATORY CHECKPOINTS	16	5	5	5	0	15	10
CORE SIGNALLING PATHWAYS	144	58	21	21	1	102	58
IMMUNOSUPPRESSIVE CYTOKINE PATHWAYS	163	57	34	33	4	122	82
Natural killers	567	249	53	42	14	377	309
IMMUNOSTIMULATORY CYTOKINES PATHWAYS	107	46	18	15	5	81	89
CORE SIGNALLING PATHWAYS	125	71	5	6	0	140	131
IMMUNOSUPPRESSIVE CYTOKINE PATHWAYS	61	21	14	6	10	38	48
NK INHIBITING RECEPTORS	48	26	2	2	1	31	72
NK ACTIVATING RECEPTORS	124	66	7	8	3	63	142
LYtic GRANULES EXOCYTOSIS	54	34	5	5	5	52	45
Dendritic cells	491	226	43	44	1	346	278
IMMUNOSTIMULATORY CYTOKINES PATHWAYS	132	66	20	21	0	89	125
ANTIGEN PRESENTATION	95	54	5	5	0	81	67
CORE SIGNALLING PATHWAYS	62	33	6	6	1	39	31
IMMUNOSUPPRESSIVE CHECKPOINTS	7	6	0	0	0	6	12
MARKERS DC	10	9	0	0	0	11	12
IMMUNOSUPPRESSIVE CYTOKINE PATHWAYS	58	28	8	8	0	37	52
RECRUITMENT OF IMMUNE CELLS	27	15	4	4	0	23	11
TUMOR RECOGNITION TUMOR KILLING	54	26	2	2	0	37	39

Supplementary Table 4

INNATE DB

Chemokine Signaling Pathway (Human)
Cytosolic DNA-sensing Pathway (Human)
Jak-STAT Signaling Pathway (Human)
MAPK Signaling Pathway (Human)
mTOR Signaling Pathway (Human)
Natural killer cell mediated cytotoxicity (Human)

KEGG -name

	ID
Toll-like receptor signaling pathway	hsa04620
Cytosolic DNA-sensing pathway	hsa04623
Natural killer cell mediated cytotoxicity	hsa04650
Antigen processing and presentation	hsa04612
Fc epsilon RI signaling pathway	hsa04664
Fc gamma R-mediated phagocytosis	hsa04666
Chemokine signaling pathway	hsa04062
Leukocyte transendothelial migration	hsa04670

REACTOME

	ID
Innate Immune System	R-HSA-168249
Interferon Signaling	R-HSA-913531.1
Signaling by Interleukins	R-HSA-449147.7
TNFR2 non-canonical NF-kB pathway	R-HSA-5668541.2
Class I MHC mediated antigen processing & presentation	R-HSA-983169.3
MHC class II antigen presentation	R-HSA-2132295.3

{}

Part III

Discussion

Chapter 9

Discussion

There are known knowns. These are things we know that we know. There are known unknowns. That is to say, there are things that we know we don't know. But there are also unknown unknowns. There are things we don't know we don't know. Donald Rumsfeld

In this work, I aimed to combine biological and mathematical expertise in order to approach the better understanding of the TME. I approached the complexity of TME with mathematical tools applied to transcriptome data. Starting this work, three years ago, little was known about the possibility of extracting immune signals from bulk tumor transcriptomes. **In this Ph.D. project I tested the limits of detection of the immune signal from bulk transcriptomes with unsupervised methods.**

Through Chapters 3-5, I tested parameters of Independent Components Analysis to optimize it for immune signal extraction. First publication (Chapter 3) focused on developing and computing the MSTD index. This work led to a better understanding of our methods. It also resulted in helpful observations for my further deconvolution work.

The work on MSTD index triggered a change in my protocol towards *overdecomposition* of transcriptomes. With my application of this protocol to six breast cancer transcriptomic datasets, I validated a hypothesis that ICA can extract reproducible signals from breast cancer transcriptomes and that some of those signals can be consequently labeled as cell-types (Chapter 4). In my additional work, I compared ICA and NMF algorithms showing that ICA breast cancer transcriptomes decomposition results in more interpretable readout (Chapter 5). Despite this result, no categorical statement on the superiority of ICA over NMF can be made. ICA and NMF differ in mathematical approach, and usually, a direct comparison (as I did) is avoided. Different *flavors* of NMF and ICA should also be tested with real data taking into account facility of use, the speed, and interpretability. Some recent works on ICA indicate that the performance of an algorithm can be strikingly different when applied to simulated and real data [?].

It is worth mentioning that the tests were performed all in breast cancer as it is one of the most fre-

quent cancers and there are publicly available datasets with large cohorts. Later results (Chapter 7) show that immune signal extracted through ICA deconvolution from breast cancer transcriptomes are the closest to the reference profiles for all three considered cell types (T cell, B cell, and Myeloid cells) and their *detectability* is quite high. This suggests that similar studies could not be reproduced in some cancer types, the, i.e., the T-cell signal is much further from the reference profiles (or not detectable) in colorectal cancer.

In Chapter 6, I implemented the methods used in the previous studies in a well-documented tool. It was an essential part of this work to make it reproducible and freely accessible to any researcher. I demonstrated that ICA is able to separate immune cell-types in blood transcriptomes with a competitive performance.

To put it in the context, in the introduction of Chapter 2, I presented a wide array of deconvolution tools. It can be observed that the field evolves impressively fast, and competition is fierce. *It would be interesting to measure the direct impact of each of the tool on the immunobiology field and percentage of the progress in immunotherapies due to the deconvolution methods.* What can be observed, it is the number and type of citations. For most of the methods published in theory-focused journals, they are rarely cited in biological works using the method. All in all, without doubts CIBERSORT met the most significant success, not only thanks to the solid scientific basis, extensive validation, and high impact journal but probably also because of the user-friendly web interface and simplicity (from the user perspective). Even though newer tools argue its accuracy for RNA-seq data, it will probably remain the champion of the field for a long time. Which does not mean that subsequent efforts are pointless as (1) CIBERSORT is under MIT license which means any new tools based on CIBERSORT methods belong to MIT (2) there is more to explore in the TME than only cell-type abundance (3) validity of CIBERSORT (and other methods) cannot be confirmed without gold standard benchmark.

The success of CIBERSORT brings into the light another critical topic which is reproducibility crisis in research [?]. It is still common to find publications about software without code or online access provided. Moreover, even if thanks to community and publishers pressure it is less and less frequent (as demonstrated with numbers in Chapter 2), still it is not trivial to understand and reproduce published tools. Often documentation is not provided or floppy. Scripts deposited on a public repository are not conformed to any standards and are not tested on different operating systems and software versions. One answer to this problem could be Docker image technology that allows sharing a *frozen* environment where the tool can run, and that is not affected by user informatics environment. However, this does not replace substantial documentation and examples provided with realized tools. I put my best effort to make my tool easy to use providing a tutorial and an R package following good practice guidelines with a sincere hope that users will be able to reproduce my work and build on without my extensive assistance.

As I stated in the Chapter 2, there is a schematic validation framework that is followed by many authors of deconvolution tools. This framework has one crucial problem: lack of gold standard validation datasets. Without a high-dimensional collection of bulk transcriptomic samples paired with an independent measure of cell-type proportions in different solid tumors, it is not possible to objectively assess the performance of published cell-type deconvolution tools. With such a

benchmark it would be easier to make the field progress in the direction that can bring most benefits to immunobiology through trustful information.

I claimed that most of the tools published in the filed are using cell profiles available from the blood and some from cancer single cell studies. The few existing unsupervised tools were not widely applied in the context of tumor transcriptomes and, if tested, they were generally overperformed by supervised frameworks.

A crucial part of an unsupervised analysis is the data interpretation. So far most of the unsupervised methods proposed deconvolution algorithm without the facilitation of the interpretation of the resulting components. This may be a reason why supervised tools met great success. Another possibility is that the data-driven nature of the unsupervised methods, that can bring unexpected results, contrasted with the more predictable behavior of supervised methods discourage researches from experimenting with them.

This brings the discussion back to our awareness of the limits of deconvolution methods in the context of the tumor transcriptomes. Through the application of DeconICA to numerous transcriptomic datasets, I observed variable detectability depending on tumor type and a number of samples. For the tumor types, some trends can be observed and interpreted. In a case of AML which immune cell-type signals were highly correlated with the reference profiles, the samples are liquid to contrast with solid tissues. Then the differences in the decomposition of solid tumors and possible biological consequences of deviation of the cell-type specific signal from the reference to be understood. The number of samples is more of technical matter. However, it is surprising that for some datasets even a low number of samples (50-90) results in meaningful signal extraction, while for some datasets with >100 samples, the decomposition can be unsuccessful. The general trend says more tumor samples, better chance to extract meaningful signals however it is not guaranteed. As the DeconICA signal extraction is data-driven and the same reference as used for all datasets, we can learn that probably use of reference genes in some tumor types may be difficult. In xCell publication [?], the authors applied their deconvolution tool to TCGA tumor types. Their sample space tSNE visualization (Fig. 4b) show that some tumor types samples can be clearly distinguished from other (AML, KIRC, SKCM, BRCA) and for some, it is more difficult (STAD, COAD, BLCA). This fact is not directly commented on by the authors. This result partially confirms our hypothesis that tissue type can play a role in the immune signal detectability.

Deep deconvolution should also be discussed. Many authors [? , ?], [?] claimed to be able to distinguish cell subtypes from transcriptome. My results say that data-driven deconvolution is not suited to accomplish this task. As shown in Chapter ??, I dissected the functional subgroups of the NK and Macrophages. Starting with single-cell resolution, I was able to detect groups of cells within studied cell types with distinct functional phenotype. Limited by the number of cells, the number of cell states was limited to two.

In Chapter 7, I decomposed five single-cell cancer transcriptomes: Melanoma, CRC, Breast, Liver, and Head and Neck. Applying ICA decomposition, I did not detect cell-subtypes. However, I identified cell types and common signals: cell-cycle, stress, etc. This shows the limits of the blind deconvolution methods for sub-types detection. Probably the gene expression profiles of immune

cell sub-types are not strong enough to be detected in an unsupervised setup.

The overall diversity of resources should also be discussed. As mentioned in Chapter 1, TCGA is the most widely used resource in pan-cancer omics studies. Main reasons are:

- accessibility
- multilevel data for the same patient
- many cancer types
- big number of patients

Studies integrating many data sources for different cancer types are less frequent as an additional effort is necessary, and even then the removal of technical biases is not guaranteed.

In hoped to overcome this issue including datasets generated with different platforms and by different research groups. In this work, I integrated data sets of published by different authors and from different technologies RNA-seq and Microarray. TCGA data were analyzed together with a corpus of datasets shared with us by Aurélien de Reynès and publicly available data. Usually, data integration poses an essential challenge in transcriptomic studies. Different tools like Combat [?] were developed to overcome this issue. Thanks to ICA, we can get rid of most of the batch effects as they can be discovered as an independent factor. In [?], the authors detected a particular batch effect thanks to one of the non-biological components. Therefore, ICA proposes a unique working framework that allowed me to compare signals from independently produced datasets in a single study without a need for re-normalization of the data. It was possible thanks to the use of ICA that removes the significant batch effects and the fact that the extracted from different datasets ICA components are comparable.

Despite the best research effort, the pan-cancer studies are missing some critical elements: time and space dimensions.

Most studies are based on omic data of tumor biopsies, which do not allow spatial localization of gathered information. In TCGA project pathology slides are available for a subset of patients. However, it is impossible to project the cell markers on them a posteriori or obtain gene expression from a selected area. Many studies, including [? ? ? ?] demonstrated that the presence of the immune cells at the invasive margin versus tumor core, or the adjacent tertiary lymphoid structures (TLS) could have a different impact on the tumor evolution and patients response to prognosis.

It is possible to estimate the abundance of some immune cells based on pathological images and also relate the patterns of cells to the patients' survival [?]. This work can be done with an algorithm or by a pathologist. However, we cannot learn anything about the nature of the cells, cell state or functional subtype from the pathological images.

Therefore, even though most of the immunophenotyping works neglect this aspect, it is important to remember that even if we assume that we can correctly estimate cell type abundance in the sample, its impact can be confounded with spatial information.

There is a hope that with an evolution of spatial transcriptomic this gap in our knowledge will be filled in a near future.

Another crucial missing in most studies dimension is time. **The immune system is a highly dynamic system.** Immune cells secrete various molecules depending on stimulants coming from the surrounding tissue (endothelial cells, blood, and lymphatic vessels), tumor cells and other immune cells. Different stimuli can have additive, suppressive or synergistic effects, the order of stimuli can also matter [?].

The animal studies can profit from time resolution data, but it is linked with a sacrifice of the animal. In human studies, this approach cannot be considered. Thus, most of the patients' samples are a *snapshot* at some time t . With sequential biopsies, it is possible to have several time points and observe a sort of dynamic nature of TME. Unfortunately, this kind of data is not widely available for many patients, and pan-cancer multi-omics cancer immunophenotyping remains wishful thinking.

Finally, I would like to discuss the place of bulk deconvolution in a single cell transcriptomic. As I mentioned before in the introduction (Chapter 1), scRNA-seq data are bringing single-cell resolution to transcriptomics. This allows studying cell states and re-definition of cell types from gene expression perspective. With few cancer single cell datasets, the scientific community learned a lot about patients and cell heterogeneity. The single cell signatures are already used in bulk deconvolution. In my work, I showed that immune cells signature extracted from a single cell are closely related to the signature extracted from the bulk transcriptomes. Due to technical challenges, scRNA-seq is not so far applied to large patient cohorts. It was also observed that rare cell types are not trivial to capture even with single-cell technologies. Even though scRNA-seq may one day replace the bulk RNA-seq, it will not be immediate. Till then both technologies should be used to cross-confirm findings and advance the state of our knowledge.

Despite continuous advances in research, our knowledge is limited on how tumor and immune cells interact with each other and how much this ecosystem is depending on intrinsic and extrinsic factors. The interest in the TME increased significantly over the last twenty years based on the percentage of publications dedicated to the TME. This was due mostly due to the vital breakthrough of immunotherapies. **Medical advances become a motivation for many projects, mine included, to be founded and perform fundamental or applied for work on the TME.** Many fundamental questions remain unanswered or controversial in the field. For example, the cell-type definition is an open issue that leads to multiple interpretations. Also, the role of different compartments of TME is now considered as context-dependent which means that it is difficult to infer a clinical-level conclusion and prognosis with heterogeneous patients. The putative predictive/explanatory still await large cohort studies followed by independent validation

As discussed briefly in the Chapter 1, researchers produce more and more data, on different scales: from molecule specific to system level which does not always directly leads to a generation of knowledge. **Biological scientists need to join their efforts with analysts (mathematicians, physicists, engineers and computer scientists) to better exploit the available data, to generate the data in a smart way which would improve our understanding of complex biological systems.**

Thanks to a multi-level transversal analysis of available data a few recent classifications of cancers based on TME features were proposed. It remains unclear how these classifications can be brought to clinical practice in a near future and what the real impact of those studies will be on patients' survival, diagnosis, and treatment selection. The descriptive character of the immunophenotyping lack simplicity (combination of various machine-learning-derived scores and knowledge-based curation) and it is not guaranteed to be universal (applicable to different cohorts, technologies).

Possible research directions will be discussed in perspectives.

Chapter 10

Conclusions and perspectives

10.1 Conclusions

This thesis described methods and results of applying unsupervised deconvolution to bulk omic data to extract cell-type specific signals.

The first contribution of this thesis is the review of deconvolution tools, including very recent ones that illustrate the diversity of the approaches to the bulk transcriptome deconvolution problems.

The second contribution is the work on methodological aspects of ICA deconvolution. I participated in the definition of Most Stable Transcriptomic Dimension (MSTD) index, and I redefined the way to apply ICA in order to extract cell-type related signals (overdecomposition) best. I demonstrated that ICA-based signals are reproducible in breast cancer and that the interpretability of ICA is higher than *brunet* version of NMF.

The third contribution is the DeconICA method for omic data deconvolution through immune components and the R package published online. DeconICA allows detection of immune cell-type signals from tumor bulk data and quantification of their abundance. The tool is not limited to ICA-based decomposition interpretation and can be easily used with different metagene generating methods. The R package has extensive documentation and tutorials that help the user to use the method autonomously. The performance of the DeconICA was evaluated in PBMC transcriptome and concluded to obtain better performance for extraction of some of the cell-type signals than the state-of-the-art published methods.

The fourth contribution is the pan-cancer DeconICA deconvolution study in which signals from 119 datasets, of 32 tumor types, with a total of 26561 samples were analyzed. The ongoing analysis highlighted detection limits of immune cell signals in some tumor types. On the natural side, so far, I focused on T-cell signal analysis which revealed that there is a high heterogeneity of t-cell signals that could be identified. Further analyses will show if this diversity has a link with patient survival or impacts other immune populations.

Finally, I contributed to a study of heterogeneity of NK and Macrophages based on scRNA-seq transcriptomic data illustrating distinct cell states of the mentioned cell types revealed thanks to a new resource: Innate free map (of the tumor microenvironment).

10.2 Perspectives

Hopefully, the achievements and findings of the thesis will not finish with the Ph.D. project itself. Many directions can be employed to continue presented work.

In the first place, the DeconICA package can still be improved. Actual compatibility of the tool with other BSS/Matrix factorization methods should be illustrated with examples, and future adjustments can be integrated into the R package. The reference signatures (for cell types and biological processes) can be extended with new signatures, i.e., based on single-cell technology if proven to bring a better interpretation to bulk decompositions. A graphical web-based interface could be a real added value and should be realized in the near future. The applicability of DeconICA to other data types, for instance, methylome is to be demonstrated.

There is a wide array of possibilities on how the analysis of ICA-based deconvoluted immune landscape can be continued. The ways I consider to be employed before journal publication of the results are:

- incorporation of clinical and survival data (when available): test for correlations between clinical features and immune cell infiltration, compare survival of patients with high and low infiltrate of different immune cell types
- better study signal reproducibility in different tumor types, better understand why in some tumor types extracted cell-type signals are closer to the reference than others.
- analysis of the diversity of Myeloid cells, B cells, CAFs, mast cells
- study of the relationship of immune signatures and cell cycle using bulk and single cell data

In a long-term perspective, the possible biological findings resulting from this work, concerning a gene or a set of genes, that would novel in the cancer immunity context could be validated *in vitro* through our partnership with the team of Vassili Soumelis.

From the more general point of view, this work could be extended in a multi-omic manner. Many groups proposed ways to combine multilevel data. Would the analysis of the immune infiltrates be more meaningful if other data types were used simultaneously?

The primary constraints for all algorithms applied to biological data are the amount of data (efficiency of the algorithm) and the course of dimensionality (large p, small n). Different data types can have specific difficulties (sparsity, missing values, drop out). Therefore in the multi-omics integration, one needs to cope with all the constraints of different data types simultaneously and the integration problem itself.

One possibility is to employ the tensor decomposition that allows simultaneous decomposition of multidimensional matrices [? ?] (called orders in tensors jargon). *Late integration* can also be considered: applying algorithms to multi-omic data independently and integrate them *a posteriori*, through a consensus [?]. Many methods were developed for multi-omics integration [?], little literature is available on the profit of multi-omics integration on the extraction of immune-related signals from bulk cancer data.

A significant constraint of unsupervised approaches is the need to use data including high variability and therefore, many samples. In theory, it could be possible to compute values for a new single sample given a space established by other samples. In practice, the values predicted for the new samples should be carefully verified for possible biases.

Finally, the blind deconvolution approaches can be applied to detect different signals from diverse tissues. For their interpretation adequate reference profiles or known signatures are necessary. Also for single-cell data, blind deconvolution can be a powerful tool to unveil new cell states.

Annexes

1 DeconICA documentation

Tutorials and manual are available as a part of R package documentation (Vignettes and Reference Manual) at <https://github.com/UrszulaCzerwinska/DeconICA>.

1.1 Introduction to deconICA

See the online tutorial *Introduction to deconICA* at: https://urszulaczerwinska.github.io/DeconICA/DeconICA_introduction.html

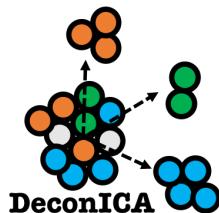
Introduction to deconICA

Deconvolution of transcriptome through Immune Component Analysis

Urszula Czerwinska

2018-05-03

- Background
 - Blind source separation
 - Application of BSS to biological data
 - Independent Components Analysis
 - NMF
 - Convex hull methods
 - Attractor metagenes
- Tutorial
 - How to install
 - Demonstration of DeconICA package
 - Simulated data
 - *in vitro* mixtures of immune cells
 - Blood data paired with FACS estimated proportions
 - Overview of functions
 - Full pipeline example
 - References



This is an introduction to the `deconICA` R package.

DeconICA stands for **D**econvolution of transcriptome through **I**mmune **C**omponent **A**nalysis.

The aim of the project is to adapt blind source separation techniques to extract immune-related signals from mixed biological samples. A great example of mixed biological sample is transcriptome measured in heterogenous tissue such as blood or tumor biopsy.

In this vignette we present short introduction to the blind source separation techniques, the biological foundation of the problem and finally we walk you through examples on how to use `deconICA` R package.

If you are interested only in practical examples of `deconICA`, skip directly to **Tutorial** section.

You can access this documentation on the [DeconICA website](#).

Background

Blind source separation

Blind source separation (BSS) is the separation of a set of source signals from a set of mixed signals, without the aid of information (or with very little information) about the source signals or the mixing process. The separation is possible under a variety of conditions.

A known example of BSS is a **cocktail party problem**, there is a group of people talking at the same time. You have multiple microphones picking up mixed signals, but you want to isolate the speech of a single person. BSS can be used to separate the individual sources by using mixed signals (Hyvärinen and Oja 2000)

Application of BSS to biological data

Through decomposition of the transcriptome matrix into components (aka factors or sources) we hope to recover underlying biological functions and cell types.

In tumor biopsies it is expected to find a part of Tumor Microenvironment (TME). TME includes tumor cells, fibroblasts, and a diversity of immune cells. Most studies have focused on individual cell types in model tumor systems, and/or on individual molecules mediating a crosstalk between two cells. Unraveling the complexity, organization, and mutual interactions of TME cellular components represents a major challenge.

Several methods have been proposed to estimate the mixing proportions of sources in biological mixtures, such as: least squares regression (Abbas et al. 2009) and more recently, non-negative least squares regression [Qiao2012], quadratic programming [Gong2011] and supported vector regression (Newman et al. 2015). Even though (Vallania et al. 2017) shows that the used algorithm do not impact substantially the results. According to Vallania et al. (2017), what matters are the gene signatures used as an input of aforementioned methods.

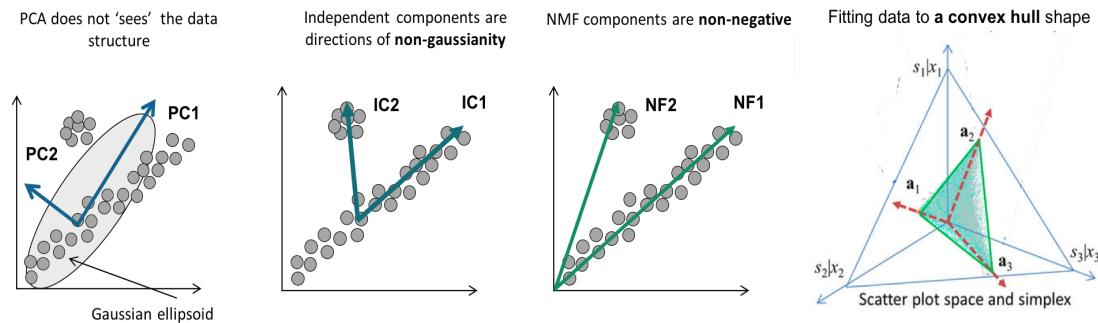
BSS methods do not use pre-defined cell-type signatures. The transcriptomic matrix is decomposed into a certain number of sources and then the sources are interpreted with available knowledge (gene signatures, cell profiles).

The main argument of using BSS over supervised decomposition techniques is that the obtained result is **unbiased** by *a priori* biological hypothesis (however there are always statistical hypothesis about the nature of data) or knowledge. In addition BSS techniques allow **discovery** of new biological signatures that can extend our available knowledge.

In the case of cell type separation from mix of tumor bulk, supervised techniques as CIBERSORT (Newman et al. 2015), MCP counter (Becht et al. 2016, Becht and de Reynies (2016)), TIMER (Li et al. 2016) etc. are based on optimized blood signatures. With an evidence brought by single cell data, these signatures are not always characterizing immune cells infiltrating tumors (Schelker et al. 2017). Some methods, like EPIC (Racle et al. 2017), use single-cell based signatures. However, today, the single cell based signatures are limited to few cancer subtypes and often based on small number of patients, incomparable with the heterogeneity that is hidden in the bulk transcriptome cohort studies.

Therefore, obtaining informative cell-type signature of immune cells infiltrating tumor biopsy samples at high throughput remains an open question that we attempt to approach with `deconICA` pipeline.

Here is a short overview of BSS or related algorithms that one can potentially use as an input to `deconICA`. At its actual state `deconICA` facilitates starting pipeline with ICA.



Graphical representation of dimension reduction & BSS methods. PCA, ICA, NMF inspired by figures of Andrei Zinovyev, Convex hull: CC BY (Wang et al. 2016)

Independent Components Analysis

Independent Component Analysis (ICA) is a matrix factorization method for data dimension reduction (Hyvärinen and Oja 2000). ICA defines a new coordinate system in the multi-dimensional space such that the distributions of the data point projections on the new axes become as mutually independent as possible. To achieve this, the standard approach is maximizing the non-gaussianity of the data point projection distributions (Hyvärinen and Oja 2000). There is no constraint imposed on the non-negativity (in contrary to

NMF) or orthogonality (in contrast to PCA). In our analysis, the negative projections are interpreted in terms of absolute values and only one side of a component is taken into account.

A mathematical way to formalize ICA is the set of equations:

the set of individual source signals $s(t) = (s_1(t), \dots, s_n(t))^T$ is mixed using a matrix $A = [a_{ij}] \in \mathbb{R}^{m \times n}$ to produce a set of mixed signals, $x(t) = (x_1(t), \dots, x_m(t))^T$, as follows:

$$x(t) = A \times s(t)$$

The above equation is effectively ‘inverted’ as follows. Blind source separation separates the set of mixed signals $x(t)$, through the determination of an ‘unmixing’ matrix to $B = [b_{ij}] \in \mathbb{R}^{m \times n}$ ‘recover’ an approximation of the original signals, $y(t) = (y_1(t), \dots, y_n(t))^T$.

$$y(t) = B \times x(t)$$

This algorithm uses higher-order moments for matrix approximation, considering all Gaussian signals as noise.

Most efficient application of ICA is fastICA (Hyvärinen and Oja 2000). However, the speed comes with a price, the results of the algorithms are not exact. This is why we recommend use of ICA with stabilization (ICASSO (Himberg and Hyvärinen 2003)) for reproducible results. More about this is the vignette [Running fastICA with icasso stabilisation](#).

For applications in molecular biology, Independent Component Analysis (ICA) models gene expression data as an action of a set of statistically independent hidden factors.

Here is a small list of ICA application to biological data:

- Independent component analysis uncovers the landscape of the bladder tumor transcriptome and reveals insights into luminal and basal subtypes (Biton et al. 2014)
- Elucidating the altered transcriptional programs in breast cancer using independent component analysis (Teschendorff et al. 2007)
- Principal Manifolds for Data Visualization and Dimension Reduction (Gorban, A.N., Kégl, B., Wunsch, D.C., Zinovyev 2008)
- Independent component analysis of microarray data in the study of endometrial cancer (Saidi et al. 2004)
- Blind source separation methods for deconvolution of complex signals in cancer biology (Zinovyev et al. 2013)
- Determining the optimal number of independent components for reproducible transcriptomic data analysis (Kairov et al. 2017)
- Application of Independent Component Analysis to Tumor Transcriptomes Reveals Specific And Reproducible Immune-related Signals (Czerwinska et al. 2018)

NMF

Non-negative matrix factorization (NMF) is matrix factorization technique assuming that the mixing, source and mixed matrices are all non negative. It is usually written as

$$V = WH$$

Matrix multiplication can be implemented as computing the column vectors of V as linear combinations of the column vectors in W using coefficients supplied by columns of H . That is, each column of V can be computed as follows:

$$v_i = Wh_i$$

where v_i is the i -th column vector of the product matrix V and h_i is the i -th column vector of the matrix H .

There are many types of NMF, that differ in implementation, ways the error terms are defined and minimized.

Gaujoux and Seoighe (2012) proposed a framework of semi-supervised NMF for solving cell and tissue mixtures in his R package *CellMix* (Gaujoux and Seoighe 2013). His work deconvolution was of great

inspiration for us and food for thoughts on the advantages and limits of BBS applied for cell type deconvolution.

Renaud Gaujoux is also an author of NMF R package (Gaujoux and Seoighe 2010; Gaujoux and Seoighe 2015b; Gaujoux and Seoighe 2015a).

Work of Cantini et al. (???) showed that ICA produces more reproducible results than NMF when applied to tumor transcriptomes.

However, we can always imagine the cases where NMF factorisation will be judged more adequate than ICA. Our pipeline is adaptable to interpretation of NMF factors without a need for major adjustments. We will be seen extend vignettes to show application of deconICA for interpretation of NMF components.

Here is a small list of NMF application to biological data:

- Metagenes and molecular pattern discovery using matrix factorization (Brunet et al. 2004)
- Tumor Clustering Using Nonnegative Matrix Factorization With Gene Selection (Chun-Hou Zheng et al. 2009)
- Semi-supervised Nonnegative Matrix Factorization for gene expression deconvolution: a case study (Gaujoux and Seoighe 2012)
- Post-modified non-negative matrix factorization for deconvoluting the gene expression profiles of specific cell types from heterogeneous clinical samples based on RNA-sequencing data (Liu et al. 2017)
- Virtual microdissection identifies distinct tumor- and stroma-specific subtypes of pancreatic ductal adenocarcinoma (Moffitt et al. 2015)
- A non-negative matrix factorization method for detecting modules in heterogeneous omics multi-modal data (Yang and Michailidis 2015)

Convex hull methods

An emerging family of BSS methods are convex geometry (CG)-based methods. Here, the “sources” are found by searching the facets of the convex hull spanned by the mapped observations solving a classical convex optimization problem (Yang and Michailidis 2015). The convex hull-based method does not require the independence assumption, nor the non-correlation assumption which can be interesting in the setup of closely related cell types. Wang et al. (2016) apply their method of convex analysis of mixtures (CAM) to tissue and cell mixtures claiming to provide new signatures. So far the published R-Java package does not allow to extract those signatures and it is not scalable to tumor transcriptomes. Another tool CellDistinguisher (Newberg et al. 2018) provides an [user-friendly R package](#). However, authors do not provide any method for estimation of number of sources. Additionally, quantitative weights are provided only for signature genes which number can vary for different sources. They do not apply their algorithm to complex mixtures as tumor transcriptome.

However, combining convex hull methods and `deconICA` can possibly lead to a meaningful interpretation.

Here is a small list of convex-hull application to biological data:

- Computational de novo discovery of distinguishing genes for biological processes and cell types in complex tissues [Newberg2018]
- Mathematical modelling of transcriptional heterogeneity identifies novel markers and subpopulations in complex tissues (Wang et al. 2016)
- Geometry of the Gene Expression Space of Individual Cells (Yang and Michailidis 2015)
- Applying unmixing to gene expression data for tumor phylogeny inference (Schwartz and Shackney 2010)
- Inferring biological tasks using Pareto analysis of high-dimensional data (Hart et al. 2015)

Attractor metagenes

Another way of generating signatures, that can be run in semi-supervised or unsupervised mode is attractor metagenes method proposed by Cheng, Ou Yang, and Anastassiou (2013). Authors describe their rationale as follows:

We can first define a consensus metagene from the average expression levels of all genes in the cluster, and rank all the individual genes in terms of their association (defined

numerically by some form of correlation) with that metagene. We can then replace the member genes of the cluster with an equal number of the top-ranked genes. Some of the original genes may naturally remain as members of the cluster, but some may be replaced, as this process will “attract” some other genes that are more strongly correlated with the cluster. We can now define a new metagene defined by the average expression levels of the genes in the newly defined cluster, and re-rank all the individual genes in terms of their association with that new metagene; and so on. It is intuitively reasonable to expect that this iterative process will eventually converge to a cluster that contains precisely the genes that are most associated with the metagene of the same cluster, so that any other individual genes will be less strongly associated with the metagene. We can think of this particular cluster defined by the convergence of this iterative process as an “attractor” i.e., a module of co-expressed genes to which many other gene sets with close but not identical membership will converge using the same computational methodology.

The produced signatures’ weights are non-negative. In the original paper, the generation of tumor signatures leads to three reproducible signatures among different tumor types. Typically with the essential parameter $\alpha = 5$, they discovered typically approximately 50 to 150 resulting attractors. Although, it is possible by tuning α obtain more or less signatures that would be possibly interpretable with `deconICA`.

Attractor metagenes R code is available on [Synapse portal](#).

Literature:

- Biomolecular Events in Cancer Revealed by Attractor Metagenes (Cheng, Ou Yang, and Anastassiou 2013)
- Discovering Genome-Wide Tag SNPs Based on the Mutual Information of the Variants (Elmas et al. 2016)
- Meta-analysis of the global gene expression profile of triple-negative breast cancer identifies genes for the prognostication and treatment of aggressive breast cancer (Al-Ejeh et al. 2014)

Tutorial

How to install

You can install `deconICA` from GitHub with:

```
#install.packages("devtools")
devtools::install_github("UrszulaCzerwinska/DeconICA", build_vignettes = TRUE)
```

or

```
install.packages("githubinstall")
githubinstall::githubinstall("DeconICA", build_vignettes = TRUE)
```

[TO DO] You can install the stable version from CRAN

```
install.packages('deconica', dependencies = TRUE)
```

Then load package with

```
library(deconica)
```

Demonstration of DeconICA package

Simulated data

At first, we assess an ability to estimate abundance of cell types in a synthetic cell mixture. Here we use function `simulate_gene_expression()` inspired by `CellMix::rmix` function. However, compared to `rmix`

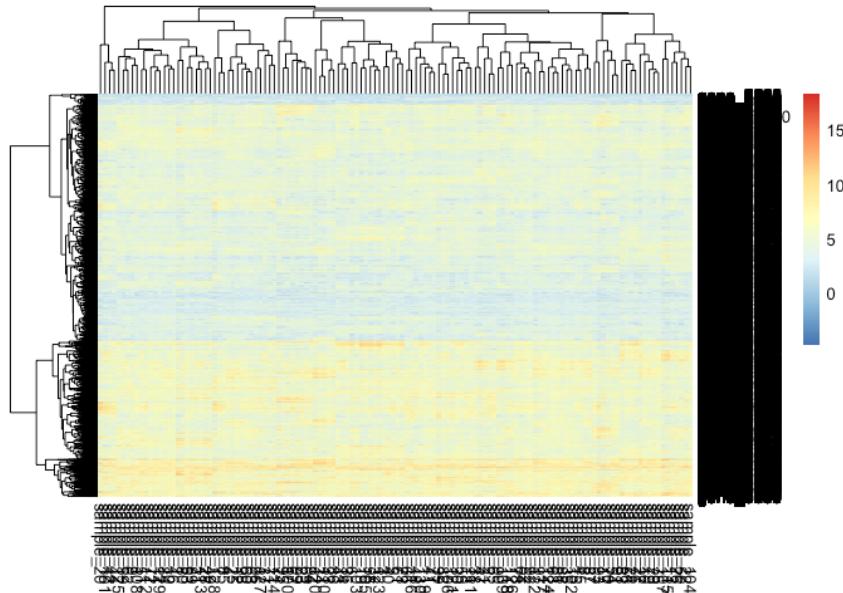
function cell profiles distribution follow user defined distribution (uniform in `rmix`) set here by default to negative binomial which approaches the biological reality.

First we create the `mix1` that is a mix of 10 cell types mixed at random proportions. Obatained matrix has 10000 genes and 130 samples. Each cell type has 20 specific markers with 2-fold difference with respect to other genes.

```
set.seed(123)
mix1<-simulate_gene_expression(10, 10000, 130, 0, markers=20)
```

We can visualize the mixed expression matrix.

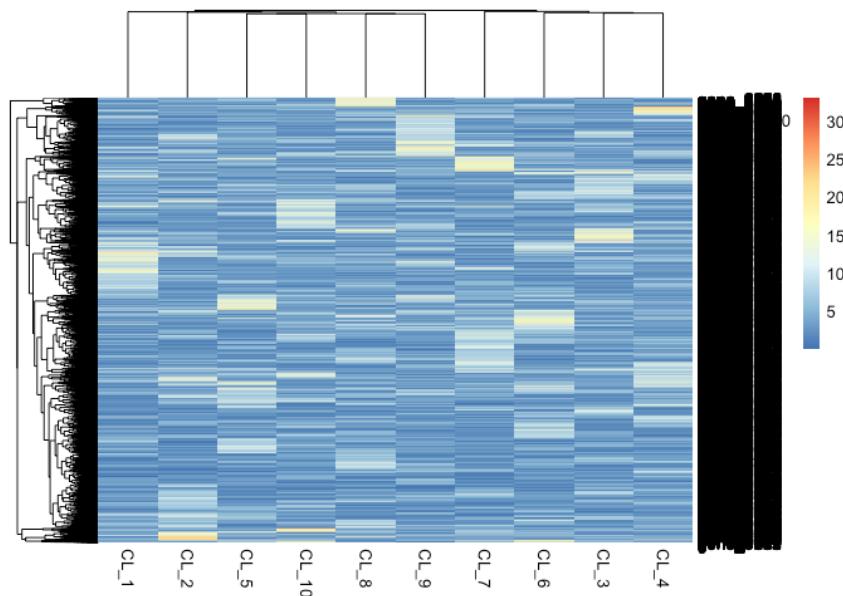
```
pheatmap::pheatmap(mix1$expression)
```



We can also visualize the

purified gene expression of each cell.

```
pheatmap::pheatmap(mix1$basis_matrix)
```



Then we apply ICA (matlab version with stabilisation see [vignette: Running fastICA with icasso stabilisation](#)) and we decompose to 11 components. If you don't have file you can find in `data-vignettes` the file `mix1_ica.RData`.

```

mix1_ica <- run_fastica (
  mix1$expression,
  overdecompose = FALSE,
  with.names = FALSE,
  gene.names = row.names(mix1$expression),
  samples = colnames(mix1$expression),
  n.comp = 11,
  R = FALSE
)

```

Subsequently, we compute correlation between components and the original cell profiles.

```

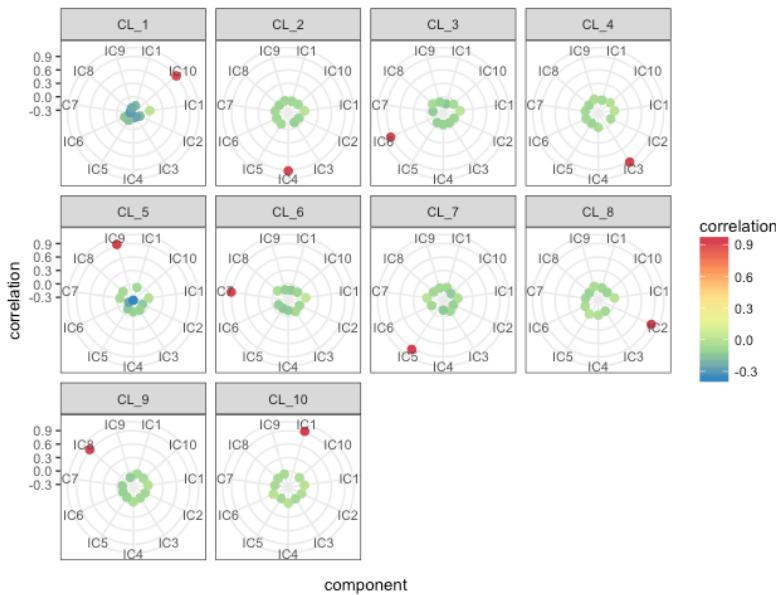
basis.list <- make_list(mix1$basis_matrix)

mix1_corr.basis <-
  correlate_metagenes (mix1_ica$S, mix1_ica$names,
                        metagenes = basis.list,
                        orient.long = TRUE,
                        orient.max = FALSE)

```

```
mix1_corr.basis_p <- radar_plot_corr(mix1_corr.basis, size.el.txt = 10, point.size = 2)
```

```
mix1_corr.basis_p$p
```



We automatically assign a component to a cell type.

```

mix1.assign <- assign_metagenes(mix1_corr.basis$r, exclude_name = NULL)
#> no profiles to exclude provided
#> DONE

```

We use top 10 genes as signatures.

```

mix1_ica.10 <-
  generate_markers(mix1_ica, 10, sel.comp = as.character(mix1.assign[, 2]))

```

It is possible to visualize *the basis matrix* as a heatmap

```

mix1_ica.10.basis <-
  generate_basis(
    df = mix1_ica,

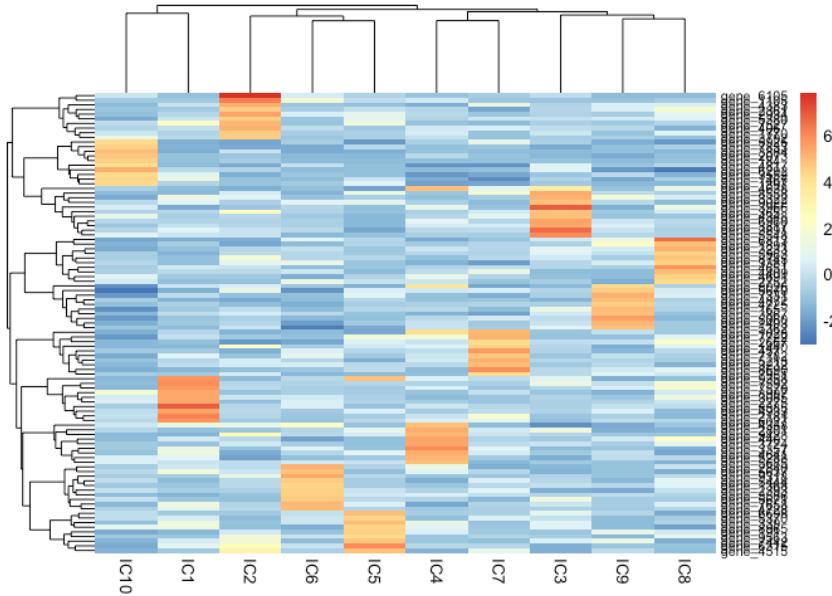
```

```

    sel.comp = as.character(mix1.assign[,2]),
    markers = mix1_ica.10
)

```

```
pheatmap::pheatmap(mix1_ica.10.basis, fontsize_row = 8)
```



We compute scores which are by default simple mean value of expression of the top genes in the original gene matrix.

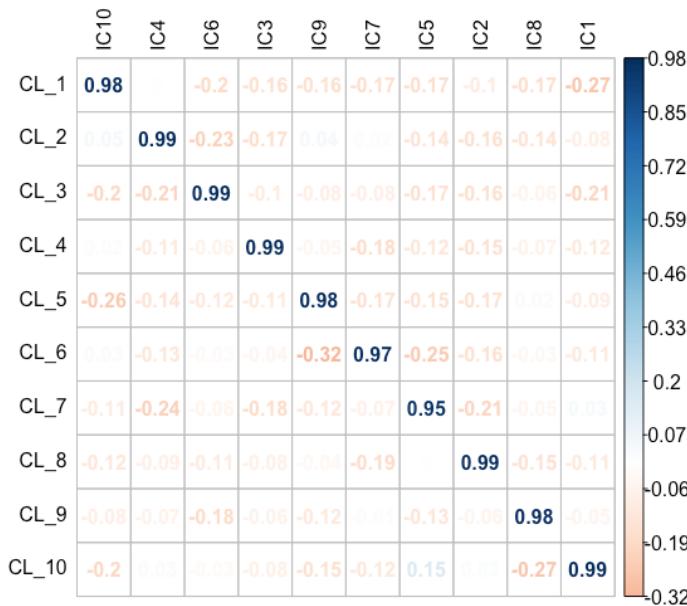
```

scores.mix1.ica <-
  get_scores(mix1_ica$log.counts, mix1_ica.10)

```

We can see on the correlation plot almost perfect correspondence with the original proportions (`mix1$prop`)

```
scores_corr_plot(scores.mix1.ica, t(mix1$prop), method="number", tl.col = "black")
```



One important feature of the ICA is that even if we do not know the exact number of components, when we overestimate the number of components, the signals are not altered. Our team proposed a method called MSTD that was developed for cancer transcriptomes to estimate optimal number of components (Kairov et al. 2017). Here we just slightly overestimate the number of components to 20.

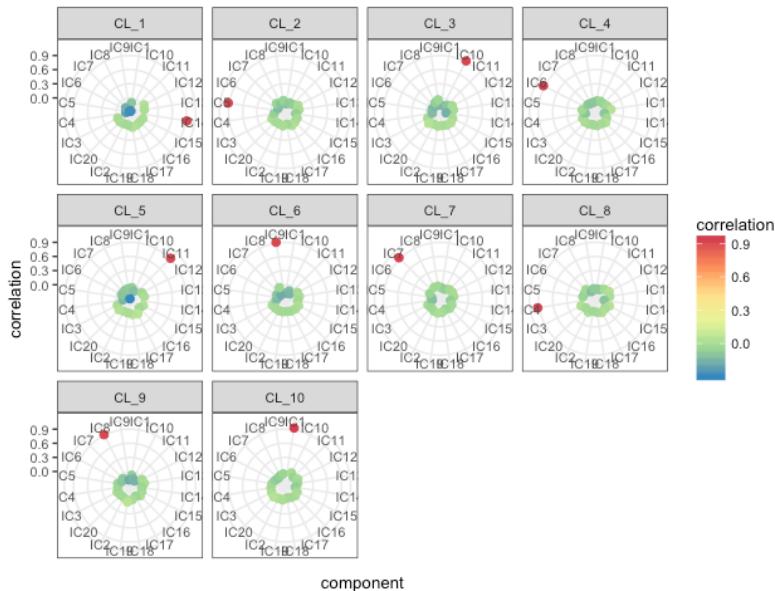
Again you can load the decomposition `mix1_ica.20.RData` from `data-vignettes` repository of the package

```
mix1_ica.20 <- run_fastica (
  mix1$expression,
  overdecompose = FALSE,
  with.names = FALSE,
  gene.names = row.names(mix1$expression),
  n.comp = 20,
  R = FALSE
)
```

Then we repeat the pipeline...

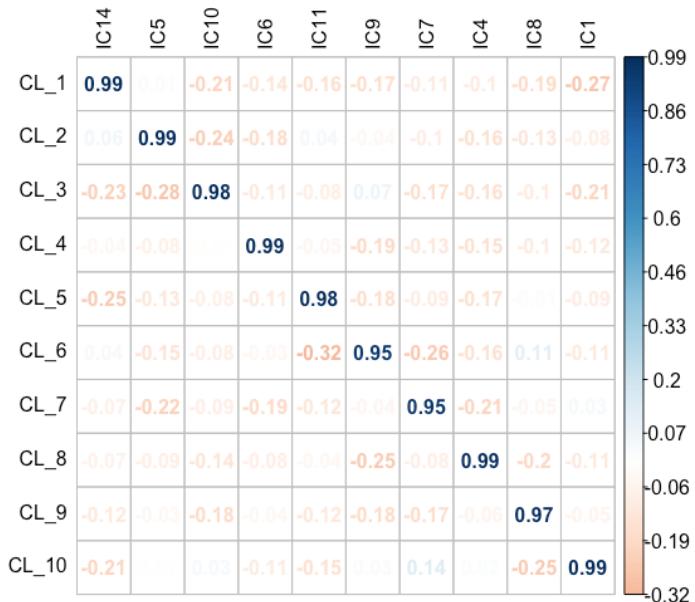
```
mix1_corr.basis.20 <-
  correlate_metagenes (
    mix1_ica.20$r,
    mix1_ica.20$names,
    metagenes = basis.list,
    orient.long = FALSE,
    orient.max = TRUE
  )
mix1_corr.basis.20_p <- radar_plot_corr(mix1_corr.basis.20,
                                         size.el.txt = 10,
                                         point.size = 2)
```

`mix1_corr.basis.20_p$p`



```
mix1.assign.20 <- assign_metagenes(mix1_corr.basis.20$r, exclude_name = NULL)
#> no profiles to exclude provided
#> DONE
mix1_ica.20$r <- mix1_corr.basis.20$S.max
mix1_ica.20_10 <- generate_markers(mix1_ica.20, 10, sel.comp = as.character(mix1.assign.20[,2]), o
scores.mix1.ica.20 <-
  get_scores(mix1_ica.20$log.counts, mix1_ica.20_10)
```

```
scores_corr_plot(scores.mix1.ica.20,t(mix1$prop), method="number", tl.col = "black")
```



And we observe that the estimated proportions are correctly estimated.

in vitro mixtures of immune cells

In this demo we use data published in ((Becht et al. 2016))[\[https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE64385\]](https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE64385) GSE64385 that you can download yourself directly from GEO using following lines of code.

```
library(Bioconductor)
library(GEOquery)
library(limma)

# Load series and platform data from GEO
GSE64385 <- getGEO("GSE64385", GSEMatrix =TRUE, AnnotGPL=TRUE)[[1]]
```

Or you can load `GSE64385.RData` from `data-vignettes` repo.

This dataset contains 5 immune cell types sorted from 3 healthy donors' peripheral bloods and mixed at different proportions.

```
head(exprs(GSE64385))
#>      GSM1570043 GSM1570044 GSM1570045 GSM1570046 GSM1570047
#> 1007_s_at   10.102138  10.155999  8.791912  8.896220  9.191570
#> 1053_at     9.228374  9.106643  7.905549  7.872256  8.001900
#> 117_at      4.965703  5.244713  9.161352  8.624783  10.440195
#> 121_at      7.096468  7.140063  6.914336  6.887061  6.759080
#> 1255_g_at    3.722354  3.781077  3.591351  3.532078  3.573182
#> 1294_at     6.785320  6.971410  9.331253  9.294259  9.130948
#>      GSM1570048 GSM1570049 GSM1570050 GSM1570051 GSM1570052
#> 1007_s_at    8.755960  8.739414  8.913141  9.065699  8.678224
#> 1053_at     7.660617  7.698507  8.052159  7.792194  7.491376
#> 117_at      9.682815  9.989239  9.024631  9.806865  10.208413
#> 121_at      6.880957  6.839934  6.845430  6.906710  6.694776
#> 1255_g_at    3.678108  3.520875  3.661857  3.572804  3.578966
#> 1294_at     9.370041  9.351603  9.296094  9.372588  9.021147
#>      GSM1570053 GSM1570054
#> 1007_s_at    8.683351  8.856388
#> 1053_at     7.742929  8.002106
#> 117_at      8.861634  9.842995
#> 121_at      7.050564  6.920022
#> 1255_g_at    3.585279  3.631184
#> 1294_at     9.417215  9.277763
```

Here is the raw matrix of proportions.

```
cell_prop <- pData(GSE64385)[ , c(1,2, 10, 11,12, 13, 14, 15, 16, 17)]
kable(cell_prop, "html", row.names = TRUE) %>%
  kable_styling(font_size = 8)
```

	title	geo_accession	characteristics_ch1	characteristics_ch1.1	characteristics_ch1.2	characteristics_ch1.3	characteristics_ch1.4	characteristics_ch1.5	characteristic
GSM1570043	Mix_1	GSM1570043	cell line: HCT116	cell line type: colon cancer	hct116 mrna mass (ng): 10	nk cells mrna mass (ng): 0	b cells mrna mass (ng): 0	neutrophils mrna mass (ng): 0	t cells mrna mass (ng): 0
GSM1570044	Mix_2	GSM1570044	cell line: HCT116	cell line type: colon cancer	hct116 mrna mass (ng): 10	nk cells mrna mass (ng): 0	b cells mrna mass (ng): 0	neutrophils mrna mass (ng): 0	t cells mrna mass (ng): 0
GSM1570045	Mix_3	GSM1570045	cell populations: HCT116, NK, B, neutrophils, T, monocytes	hct116 mrna mass (ng): 10	nk cells mrna mass (ng): 10	b cells mrna mass (ng): 0.6	neutrophils mrna mass (ng): 0.3	t cells mrna mass (ng): 2.5	monocytes mrna mass (ng): 5
GSM1570046	Mix_4	GSM1570046	cell populations: HCT116, NK, B, neutrophils, T, monocytes	hct116 mrna mass (ng): 10	nk cells mrna mass (ng): 5	b cells mrna mass (ng): 10	neutrophils mrna mass (ng): 0.2	t cells mrna mass (ng): 1.3	monocytes mrna mass (ng): 2.5
GSM1570047	Mix_5	GSM1570047	cell populations: HCT116, NK, B, neutrophils, T, monocytes	hct116 mrna mass (ng): 10	nk cells mrna mass (ng): 2.5	b cells mrna mass (ng): 5	neutrophils mrna mass (ng): 2.5	t cells mrna mass (ng): 0.6	monocytes mrna mass (ng): 1.3
GSM1570048	Mix_6	GSM1570048	cell populations: HCT116, NK, B, neutrophils, T, monocytes	hct116 mrna mass (ng): 10	nk cells mrna mass (ng): 1.3	b cells mrna mass (ng): 2.5	neutrophils mrna mass (ng): 1.3	t cells mrna mass (ng): 10	monocytes mrna mass (ng): 0.6
GSM1570049	Mix_7	GSM1570049	cell populations: HCT116, NK, B, neutrophils, T, monocytes	hct116 mrna mass (ng): 10	nk cells mrna mass (ng): 0.6	b cells mrna mass (ng): 1.3	neutrophils mrna mass (ng): 0.6	t cells mrna mass (ng): 5	monocytes mrna mass (ng): 10
GSM1570050	Mix_8	GSM1570050	cell populations: HCT116, NK, B, neutrophils, T, monocytes	hct116 mrna mass (ng): 10	nk cells mrna mass (ng): 10	b cells mrna mass (ng): 5	neutrophils mrna mass (ng): 0.6	t cells mrna mass (ng): 1.3	monocytes mrna mass (ng): 0.6
GSM1570051	Mix_9	GSM1570051	cell populations: HCT116, NK, B, neutrophils, T, monocytes	hct116 mrna mass (ng): 10	nk cells mrna mass (ng): 0.6	b cells mrna mass (ng): 10	neutrophils mrna mass (ng): 1.3	t cells mrna mass (ng): 2.5	monocytes mrna mass (ng): 1.3
GSM1570052	Mix_10	GSM1570052	cell populations: HCT116, NK, B, neutrophils, T, monocytes	hct116 mrna mass (ng): 10	nk cells mrna mass (ng): 1.3	b cells mrna mass (ng): 0.6	neutrophils mrna mass (ng): 2.5	t cells mrna mass (ng): 5	monocytes mrna mass (ng): 2.5
GSM1570053	Mix_11	GSM1570053	cell populations: HCT116, NK, B, neutrophils, T, monocytes	hct116 mrna mass (ng): 10	nk cells mrna mass (ng): 2.5	b cells mrna mass (ng): 1.3	neutrophils mrna mass (ng): 0.2	t cells mrna mass (ng): 10	monocytes mrna mass (ng): 5
GSM1570054	Mix_12	GSM1570054	cell populations: HCT116, NK, B, neutrophils, T, monocytes	hct116 mrna mass (ng): 10	nk cells mrna mass (ng): 5	b cells mrna mass (ng): 2.5	neutrophils mrna mass (ng): 0.3	t cells mrna mass (ng): 0.6	monocytes mrna mass (ng): 10

We manually extracted the immune cell proportions from the matrix

```
cell_prop.clean <- cell_prop [,1:2]
cell_prop.clean$NK <- c(0,0, 10, 5, 2.5, 1.3, 0.6, 10, 0.6, 1.3, 2.5, 5)
cell_prop.clean$Bcell <- c(0,0, 0.6, 10, 5, 2.5, 1.3, 5, 10, 0.6, 1.3, 2.5)
cell_prop.clean$Neutrophils <- c(0,0, 0.3, 0.2, 2.5, 1.3, 0.6, 0.6, 0.6, 1.3, 2.5, 0.2, 0.3 )
cell_prop.clean$Tcell <- c(0,0,2.5, 1.3, 0.6, 10, 5, 1.3, 2.5, 5,10,0.6 )
cell_prop.clean$Monocytes <- c(0,0, 5, 2.5, 1.3, 0.6, 10, 0.6, 1.3, 2.5, 5, 10)

kable(cell_prop.clean, "html", row.names = TRUE) %>%
  kable_styling(font_size = 8)
```

	title	geo_accession	NK	Bcell	Neutrophils	Tcell	Monocytes
GSM1570043	Mix_1	GSM1570043	0.0	0.0	0.0	0.0	0.0
GSM1570044	Mix_2	GSM1570044	0.0	0.0	0.0	0.0	0.0
GSM1570045	Mix_3	GSM1570045	10.0	0.6	0.3	2.5	5.0
GSM1570046	Mix_4	GSM1570046	5.0	10.0	0.2	1.3	2.5
GSM1570047	Mix_5	GSM1570047	2.5	5.0	2.5	0.6	1.3
GSM1570048	Mix_6	GSM1570048	1.3	2.5	1.3	10.0	0.6
GSM1570049	Mix_7	GSM1570049	0.6	1.3	0.6	5.0	10.0
GSM1570050	Mix_8	GSM1570050	10.0	5.0	0.6	1.3	0.6
GSM1570051	Mix_9	GSM1570051	0.6	10.0	1.3	2.5	1.3
GSM1570052	Mix_10	GSM1570052	1.3	0.6	2.5	5.0	2.5

	title	geo_accession	NK	Bcell	Neutrophils	Tcell	Monocytes
GSM1570053	Mix_11	GSM1570053	2.5	1.3	0.2	10.0	5.0
GSM1570054	Mix_12	GSM1570054	5.0	2.5	0.3	0.6	10.0

Then we scaled them to relative proportions.

```
rowSums(cell_prop.clean[, 3:7])
#> GSM1570043 GSM1570044 GSM1570045 GSM1570046 GSM1570047 GSM1570048
#>     0.0      0.0     18.4     19.0     11.9     15.7
#> GSM1570049 GSM1570050 GSM1570051 GSM1570052 GSM1570053 GSM1570054
#>     17.5     17.5    15.7     11.9     19.0     18.4
cell_prop.clean.scaled <- 
  cell_prop.clean[, 3:7] / rowSums(cell_prop.clean[, 3:7])
kable(cell_prop.clean.scaled, "html", row.names = TRUE) %>%
  kable_styling(font_size = 8)
```

	NK	Bcell	Neutrophils	Tcell	Monocytes
GSM1570043	NaN	NaN	NaN	NaN	NaN
GSM1570044	NaN	NaN	NaN	NaN	NaN
GSM1570045	0.5434783	0.0326087	0.0163043	0.1358696	0.2717391
GSM1570046	0.2631579	0.5263158	0.0105263	0.0684211	0.1315789
GSM1570047	0.2100840	0.4201681	0.2100840	0.0504202	0.1092437
GSM1570048	0.0828025	0.1592357	0.0828025	0.6369427	0.0382166
GSM1570049	0.0342857	0.0742857	0.0342857	0.2857143	0.5714286
GSM1570050	0.5714286	0.2857143	0.0342857	0.0742857	0.0342857
GSM1570051	0.0382166	0.6369427	0.0828025	0.1592357	0.0828025
GSM1570052	0.1092437	0.0504202	0.2100840	0.4201681	0.2100840
GSM1570053	0.1315789	0.0684211	0.0105263	0.5263158	0.2631579
GSM1570054	0.2717391	0.1358696	0.0163043	0.0326087	0.5434783

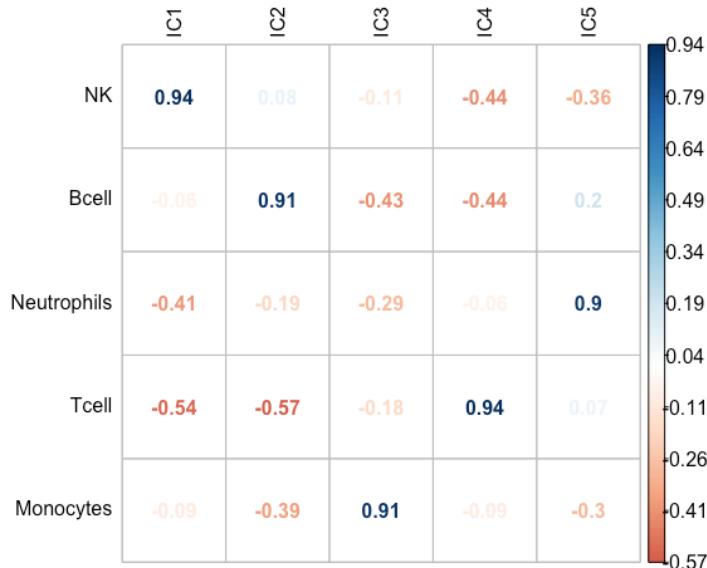
Then we performed ICA decomposing into 6 components as we account for the possible junk component.

```
GSE64385_ica <- run_fastica (
  exprs(GSE64385),
  overdecompose = FALSE,
  with.names = FALSE,
  gene.names = row.names(exprs(GSE64385)),
  samples = colnames(exprs(GSE64385)),
  n.comp = 6,
  R = FALSE
)
```

The components are oriented according to *long tail* even without verifying correlations, once we select the top 10 genes of each component we can generate scores.

```
GSE64385_ica_markers_10 <- generate_markers(GSE64385_ica, 10)
GSE64385.ica_scores <-
  get_scores(GSE64385_ica$log.counts, GSE64385_ica_markers_10)

scores_corr_plot(GSE64385.ica_scores[,1:5], cell_prop.clean.scaled, method="number", t1.col = "blac
```



And even better accuracy can be obtained if we compute scores on *un-log*ged data, actual counts.

```
GSE64385.ica_scores_unlog <-
  get_scores((^GSE64385_ica$log.counts)-1, GSE64385_ica_markers_10)
```

```
scores_corr_plot(GSE64385.ica_scores_unlog[,1:5],cell_prop.clean.scaled[1:5], method="number", t1.
```



Blood data paired with FACS estimated proportions

Here we will use `SDY420` dataset from `Immport` database that Aran, Hu, and Butte (2017) kindly shared with us. They contain PBMC expression data of 104 healthy patients and paired FACS proportion estimation.

```
#import expression data
GE_SDY420 <- read.delim("./data-raw/xCell_ImmPort/GE_SDY420.txt", row.names=1, stringsAsFactors=FALS
```

```
dim(GE_SDY420)
#> [1] 12027   104
```

```

summary(GE_SDY420)[,1:3]
#>   SUB137169      SUB137172      SUB137208
#> Min. : 5.840  Min. : 4.099  Min. : 4.140
#> 1st Qu.: 6.254 1st Qu.: 6.211 1st Qu.: 6.083
#> Median : 6.716 Median : 6.742 Median : 7.137
#> Mean   : 7.335 Mean   : 7.335 Mean   : 7.336
#> 3rd Qu.: 7.982 3rd Qu.: 8.041 3rd Qu.: 8.278
#> Max.   :15.059 Max.   :15.664 Max.   :15.216

```

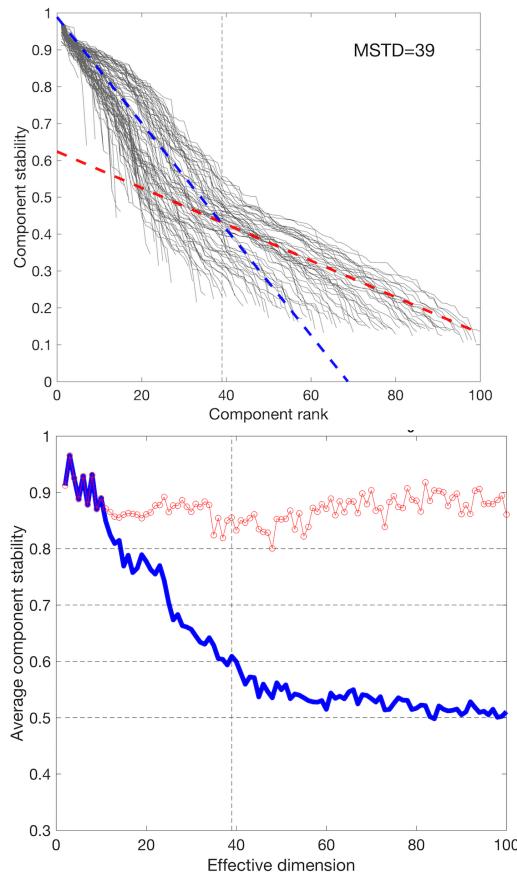
In order to define number of components we can perform `doICABatch` that scans an array of decompositions looking for the most reproducible ones. The methodology was published in (Kairov et al. 2017).

The function saves on the disk all the studied decomposition and generates plots of stability.

```

GE_SDY420_batch.res<-doICABatch(GE_SDY420,
                                    seq(2,100,1),
                                    names = row.names(GE_SDY420),
                                    samples = colnames(GE_SDY420))

```



The MST = 39 indicates most reproducible number of components. Let's verify if among 39 components we find components associated with the immune cells.

You can load the file with decomposition from `data-vignettes` repo.

```

GE_SDY420_ica_39 <- run_fastica (
  GE_SDY420,
  gene.names = row.names(GE_SDY420),
  samples = colnames(GE_SDY420),
  overdecompose = FALSE,
  with.names = FALSE,
  n.comp = 39,
)

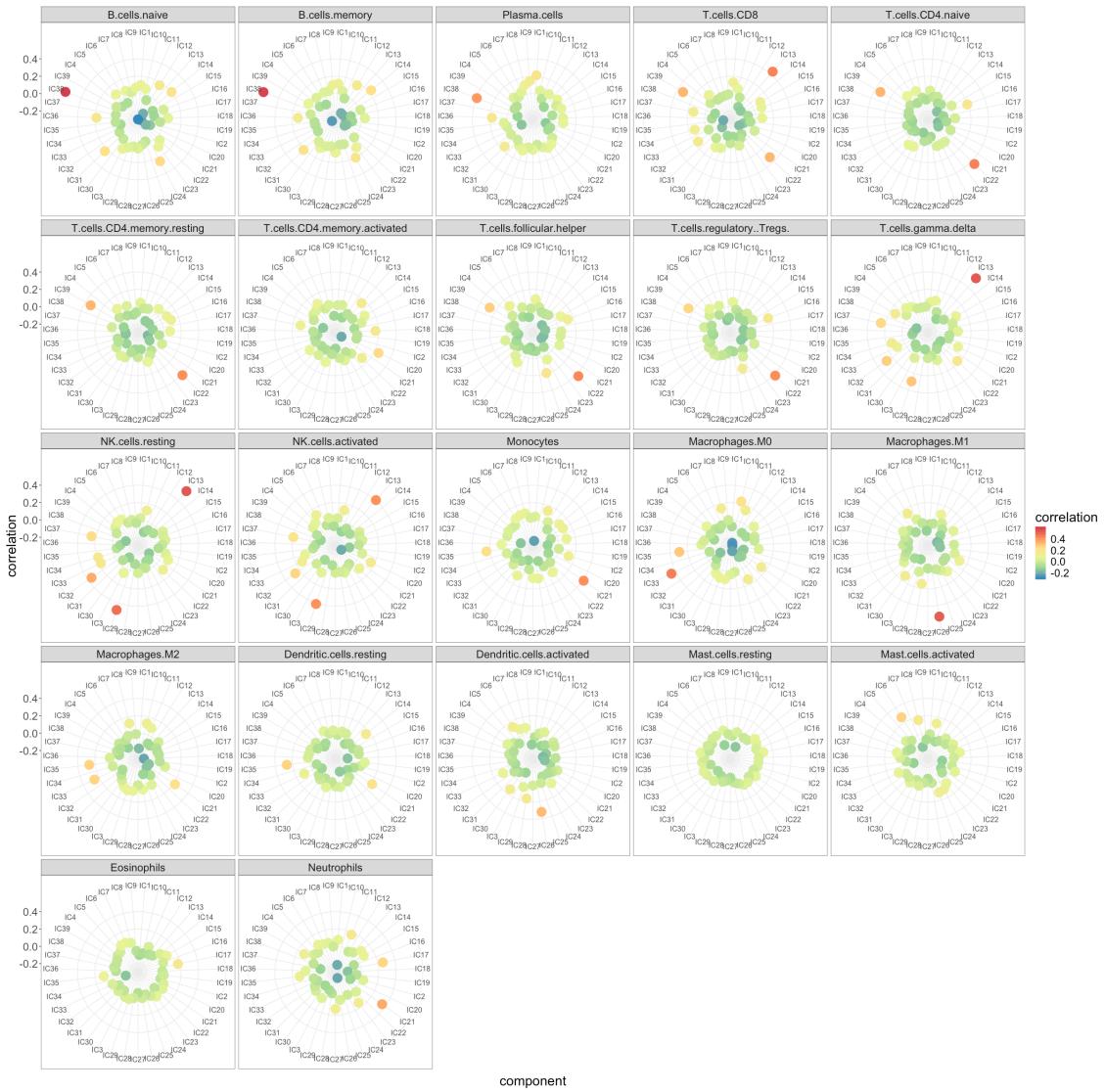
```

```
R = FALSE  
)
```

Then we perform our pipeline comparing to pure immune profiles from (Newman et al. 2015).

```
GE_SDY420_ica_39.corr.LM22 <-  
  correlate_metagenes (GE_SDY420_ica_39$S, GE_SDY420_ica_39$names, metagenes = LM22.list)  
  
GE_SDY420_ica_39.corr.LM22.p <-  
  radar_plot_corr(GE_SDY420_ica_39.corr.LM22,  
    ax.size = 12,  
    size.el.txt = 22,  
    point.size = 7)
```

```
GE_SDY420_ica_39.corr.LM22.p$p
```



```
GE_SDY420_ica_39.LM22.reciprocal.corr <-  
  assign_metagenes(GE_SDY420_ica_39.corr.LM22$r, exclude_name = NULL)  
#> no profiles to exclude provided  
#> DONE
```

```
kable(GE_SDY420_ica_39.LM22.reciprocal.corr, "html", row.names = FALSE)
```

profile

B.cells.naive

component

IC38

profile	component
T.cells.CD4.naive	IC22
T.cells.CD4.memory.activated	IC20
NK.cells.resting	IC13
Monocytes	IC21
Macrophages.M0	IC33
Macrophages.M1	IC26
Mast.cells.activated	IC6

```
GE_SDY420_ica_39.LM22.max.corr <- get_max_correlations(GE_SDY420_ica_39.corr.LM22)
```

```
kable(GE_SDY420_ica_39.LM22.max.corr, "html", row.names = FALSE)
```

TYPE	IC	r	p.val
B.cells.naive	IC38	0.5965356	0.0000000
B.cells.memory	IC38	0.5892516	0.0000000
Plasma.cells	IC38	0.3934263	0.0000000
T.cells.CD8	IC13	0.4378895	0.0000000
T.cells.CD4.naive	IC22	0.4459369	0.0000000
T.cells.CD4.memory.resting	IC22	0.4117528	0.0000000
T.cells.CD4.memory.activated	IC20	0.2510106	0.0000035
T.cells.follicular.helper	IC22	0.4280154	0.0000000
T.cells.regulatory..Tregs.	IC22	0.4187166	0.0000000
T.cells.gamma.delta	IC13	0.5383511	0.0000000
NK.cells.resting	IC13	0.5429902	0.0000000
NK.cells.activated	IC29	0.4101813	0.0000000
Monocytes	IC21	0.4123837	0.0000000
Macrophages.M0	IC33	0.4538960	0.0000000
Macrophages.M1	IC26	0.5301073	0.0000000
Macrophages.M2	IC35	0.2675745	0.0000007
Dendritic.cells.resting	IC35	0.2671913	0.0000008
Dendritic.cells.activated	IC26	0.3215513	0.0000000
Mast.cells.resting	IC17	0.0898357	0.1017364
Mast.cells.activated	IC6	0.2701451	0.0000006
Eosinophils	IC17	0.1723216	0.0015973
Neutrophils	IC21	0.3548045	0.0000000

Both reciprocal and maximal correlations indicate a set of components that can be labelled as immune cells. Let's verify if we can use them to estimate proportions of those cell types.

```

SDY420_markers_10 <-
  generate_markers(
    df = GE_SDY420_ica_39,
    n = 10,
    sel.comp = as.character(unique(GE_SDY420_ica_39.LM22.max.corr$IC)),
    return = "gene.list"
  )

```

```

GE_SDY420_ica_39_scores <-
  get_scores ((2 ^ GE_SDY420_ica_39$log.counts) - 1,
  SDY420_markers_10,
  summary = "mean",
  na.rm = TRUE
)

```

```

head(GE_SDY420_ica_39_scores)
#>           IC38      IC13      IC22      IC20      IC29      IC21
#> SUB137169 1030.7341 1336.840 1832.286 155.9163 899.3425 1074.8662
#> SUB137172 1818.6165 1123.062 1374.354 157.4036 960.5435 2927.9662
#> SUB137208 1118.0113 1350.213 1465.132 150.7131 906.9577 1234.6859
#> SUB137209 981.7916 1427.991 1705.373 158.0823 926.9521 969.4372
#> SUB137220 1010.8262 2430.426 1655.813 179.0462 1037.9035 950.5174
#> SUB137224 794.8479 1296.308 1484.678 160.0176 1016.9566 701.2229
#>           IC33      IC26      IC35      IC17      IC6
#> SUB137169 1261.044 393.0449 604.0954 123.63160 1232.1581
#> SUB137172 1011.749 710.6023 1531.9271 69.45387 3012.6474
#> SUB137208 1687.398 523.3801 594.8693 843.23650 619.2452
#> SUB137209 1032.434 1021.3100 454.9859 183.05840 4064.6511
#> SUB137220 1351.556 436.3534 594.5121 70.55523 1287.1568
#> SUB137224 1490.662 415.4843 438.7275 116.07239 732.3497

```

As we have FACS measured proportions we can import them.

```

#import facs estimated proportions
FACS_SDY420 <- read.delim("../data-raw/xCell_ImmPort/FCS_SDY420.txt", row.names=1, stringsAsFactors=

```

```

dim(FACS_SDY420)
#> [1] 24 104

```

```

common.samples <- intersect(colnames(FACS_SDY420), colnames(GE_SDY420))
length(common.samples)
#> [1] 104

```

```

FACS_SDY420.fil <- data.frame(t(FACS_SDY420[, common.samples]))

```

```

head(FACS_SDY420.fil)[,1:4]
#>           B.cells CD16..monocytes CD16..monocytes.1 CD4..T.cells
#> SUB137169  0.0710        0.1430        0.0080        0.4138
#> SUB137172  0.1342        0.1695        0.0155        0.2014
#> SUB137208  0.1344        0.2324        0.0101        0.2301
#> SUB137209  0.0858        0.2006        0.0114        0.2718
#> SUB137220  0.0493        0.1600        0.0089        0.2618
#> SUB137224  0.0626        0.1663        0.0136        0.3320

```

And we can confront the abundance scores.

```

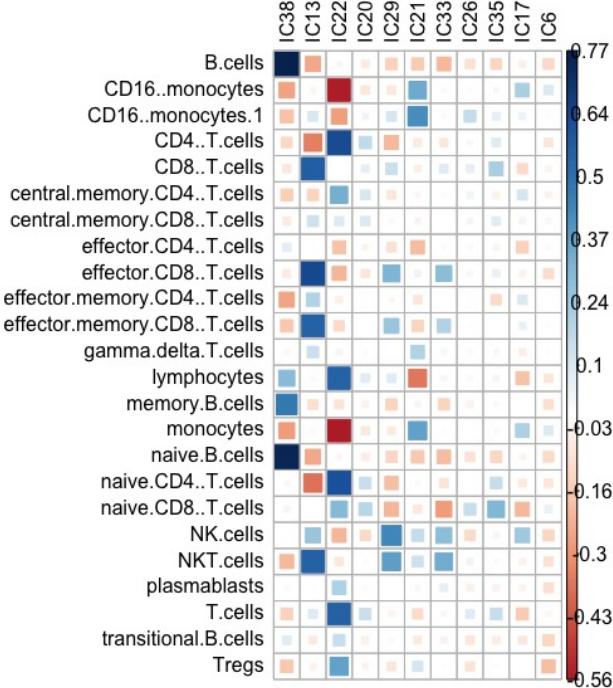
scores_corr_plot(GE_SDY420_ica_39_scores,FACS_SDY420.fil, tl.col = "black")

```

```

knitr::include_graphics("./figures-ext/CorrBlood.jpeg")

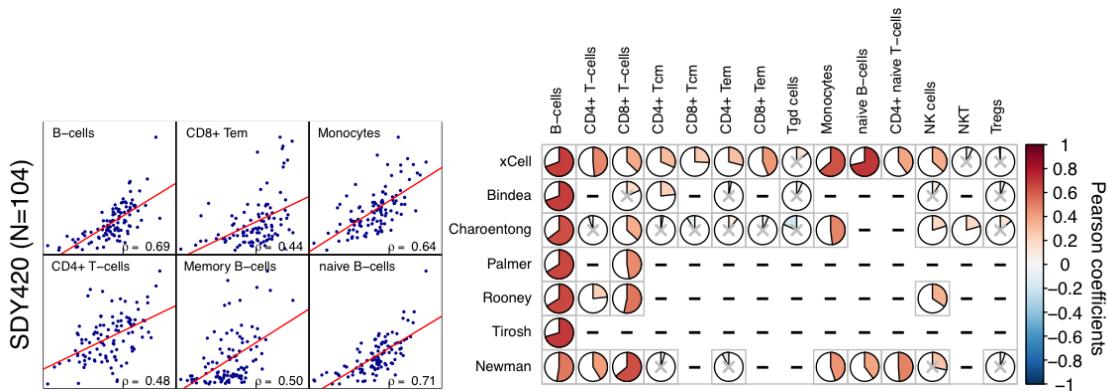
```



The estimation of abundance gives remarkable accuracy (Pearson correlation coefficient):

- IC38: B-cells: 0.76
- CD4 T-cells: 0.629
- CD8 T-cells: 0.63
- NK: 0.43
- Monocytes: 0.42

This results is also highly comparable with (or even better than) results obtained with other tools, published in *xCell* publication by Aran, Hu, and Butte (2017).



Overview of functions

In this section we will discuss main functions of the package and their different options on computationally light toy examples. If you are interested in demo application of `deconICA` with biological arguments go straight to section [Demonstration of DeconICA package](#)

You can use `run_fastica()` function to decompose a matrix into independent components

```
S <- matrix(runif(10000), 5000, 2)
A <- matrix(c(1, 1, -1, 3), 2, 2, byrow = TRUE)
X <- data.frame(S %*% A)
res <- run_fastica(X = X, row.center = TRUE, n.comp = 2, overdecompose = FALSE)
#> running PCA
#> running ICA for 2 components
#> adding names to the object
#> adding sample names to the object
```

```

#> adding counts in Log to the object
str(res)
#> List of 8
#> $ X      : num [1:5000, 1:2] 0.989 1.337 1.133 -1.105 -1.281 ...
#> $ K      : num [1:2, 1:2] -1.22 -6.69e-16 -9.93 1.81e+16
#> $ W      : num [1:2, 1:2] 0.0137 0.9999 -0.9999 0.0137
#> $ A      : num [1:2, 1:2] -Inf Inf NaN NaN
#> ... attr(*, "dimnames")=List of 2
#> ... .$. : chr [1:2] "IC1" "IC2"
#> ... .$. : chr [1:2] "X1" "X2"
#> $ S      : num [1:5000, 1:2] 0.105 -0.103 0.178 0.885 2.639 ...
#> ... attr(*, "dimnames")=List of 2
#> ... .$. : chr [1:5000] "X1" "X2" "X3" "X4" ...
#> ... .$. : chr [1:2] "IC1" "IC2"
#> $ names   : chr [1:5000] "X1" "X2" "X3" "X4" ...
#> $ samples  : chr [1:2] "X1" "X2"
#> $ Log.counts:'data.frame': 5000 obs. of 2 variables:
#> ..$. X1: num [1:5000] 0.8349 0.1645 0.885 -0.0892 -0.8865 ...
#> ..$. X2: num [1:5000] 1.425 0.262 1.272 3.462 2.914 ...

```

`run_fastica` runs `fastica` from `fastica` package. In this trivial example we create sources matrix `S` and mixing matrix `A` that we multiply to obtain `X`. Then we decompose `X` into `n.comp = 2`, with row centering (subtracting mean from each row) `row.center = TRUE`. We also checked `overdecompose = FALSE`, `overdecompose = TRUE` would ignore number of components we defined with `ncomp`. It finds its use for more advanced analysis applied to transcriptome [see section [Demonstration of DeconICA package](#)]. Other parameters were selected as default.

Full description of the `run_fastica` parameters can be found in help `?run_fastica`.

The main differences between `run_fastica` and `fastica` are:

- if column names are provided with the matrix, duplicated names are removed and entries with higher variance are kept
- it transforms data into log2 if data are in row counts
- it runs a PCA before ICA to denoise the matrix
- it allows running matlab version fastica with *icasso* stabilisation if matlab software is installed on your machine (more about this point in [vignette: Running fastICA with icasso stabilisation](#))
- it returns in a `list`
 - input `data.frame` without duplicated entries and before log transformation: `log.counts`
 - `names` row names vector
 - `samples` sample names vector
 - `A`, `S`, `K` and `W` matrices (see `?run_fastica` for details) if run in `R=TRUE`
 - `A` and `S` matrices if run with `R=FALSE`
- `overdecompose` parameter that selects number of composed needed to perform overdecomposition of the input matrix

Therefore, `run_fastica()` performs ICA decomposition of the matrix and provides additional features useful for the downstream analysis. The use of more advanced options will be demonstrated later on in this tutorial. It generates *components* or *sources* to which we will refer later on in the tutorial.

The step naturally following `run_fastica()` is `correlate_metagenes()`.

It is common that after an unsupervised decomposition, components should have attributed an interpretation or a meaning or a label. We call this process *interpret* components or *identify* sources. A domain knowledge is necessary to interpret components. In the case of transcriptomic data, components can be seen as weighted gene list.

An efficient way to interpret a component is to use correlation with some known profile or as we call it a *metagene*. If we dispose of a known weighted list of genes that characterize a biological phenomena or a cell type (a metagene), we can then correlate them with obtained components and verify if some of decomposed sources are close to the known cells/functions.

In our `deconICA` pipeline `correlate_metagenes()` can be used in order to correlate metagenes (gene lists: knowledge-based or data-driven) with the data-driven components.

```
library(deconica)
data(Example_ds)
#decompose the matrix
set.seed(123)
res_run_ica <- run_fastica (
  Example_ds,
  overdecompose = FALSE,
  with.names = TRUE,
  n.comp = 10
)
#> running PCA
#> running ICA for 10 components
#> adding names to the object
#> adding sample names to the object
#> adding counts in Log to the object

#correlate with Biton et al. metagenes
corr <- correlate_metagenes(S = res_run_ica$S,
                             gene.names = res_run_ica$names,
                             metagenes = Biton.list
                           )
```

In this case we use an example of an extract, of 60 samples, from transcriptomic data from breast cancer (Bekhouche et al. 2011). At first, dataset is decomposed into an arbitrary number of 10 components. The `correlate_metagenes()` correlates the obtained `S` matrix with *Biton et al.* metagenes (Biton et al. 2014). This set of 11 metagenes is data-driven and was derived in the article Biton et al. (2014) from pan-cancer transcriptome as the reproducible signals, common to many cancer types. Some of the signatures as BCLAPATHWAYS or UROTHELIALDIF (urothelial differentiation) were shared among many datasets, but within bladder cancer. They can be as a sort of negative control.

However, any set weighted signatures with reasonable size can be used as metagenes. Later in this tutorial we use immune cell profiles optimized for cell type deconvolution from (Newman et al. 2015) `LM22.list`.

One can control a minimal number of genes used for correlation with `n.genes.intersect` it is set to a magic number in statistics (30) by default.

```
names(corr)
#> [1] "S.or"   "n"     "r"     "P"
```

The `correlate_metagenes()` returns `n`, `r`, `P` matrices which correspond to `Hmisc::rcorr` function output, number of genes on which correlation is based, correlation coefficient and p-value.

The `S.or` stands for `S` matrix that is *oriented*. Why the matrix should be oriented? If you used ICA to decompose gene expression then the positive and negative projections do not have meaning by itself. We developed a methodology orienting data in the direction of the *long tail* of the distribution. Which means highest absolute values of a component weight should be positive. However, if the distribution doesn't have *tails* and is close to gaussian, an alternative, can be orienting components through maximal correlation. If we are confident with our metagenes, we can orient the `S` matrix so that the maximal correlation is always positive. We demonstrate the use of `orient.max = FALSE` and obtained `S.or` in [use cases section](#). One can decide not to orient the `S` matrix through selecting `orient.long = FALSE` and `orient.max = FALSE`.

If we have a look at `r` matrix, we see, indeed, it contains correlations between components and provided metagenes.

```
head(corr$r)
#>      M12_MYOFIBROBLASTS M13_BLCAPATHWAYS  M14_STRESS M2_GC_CONTENT
#> IC1      5.241598e-03      0.11551181 -0.05863261  0.001829696
#> IC2      3.872262e-02      0.05786138  0.11436405  0.006144067
#> IC3      8.789440e-02      0.02586692 -0.01506925 -0.027817853
#> IC4      7.534325e-01      -0.02125887  0.22186764  0.062138866
```

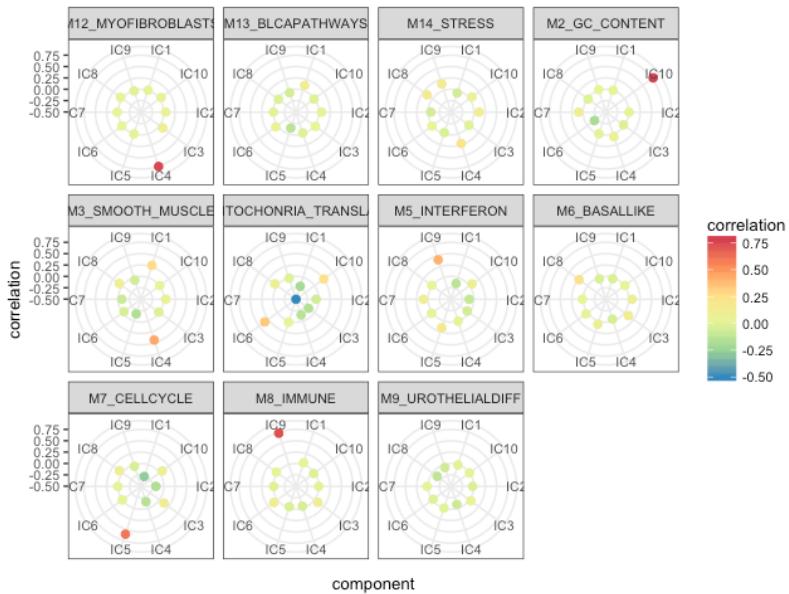
```

#> IC5      -1.393165e-05   -0.13490904  -0.02972322  -0.001664296
#> IC6      1.442049e-02    0.04582809  0.01508958  -0.189164862
#> M3_SMOOTH_MUSCLE M4_MITOCHONRIA_TRANSLATION M5_INTERFERON M6_BASALLIKE
#> IC1      0.28082149     -0.20320196  -0.138082269 -0.02895456
#> IC2      0.04300946     -0.05986897  -0.103115638 0.07220367
#> IC3      -0.01987917    -0.16167704  -0.061727219 0.10881091
#> IC4      0.44002012     -0.14025239  0.003064779 -0.04316638
#> IC5      -0.16530245    0.02677519  0.156166568  0.07481530
#> IC6      -0.04016218    0.34503586  0.023152228  0.04141703
#> M7_CELLCYCLE M8_IMMUNE M9_UROTHELIALDIFF
#> IC1      -0.275278283  0.042007208 -0.007860347
#> IC2      -0.173358489  -0.004013247 -0.026667308
#> IC3      0.123241033  0.093491330 0.037031845
#> IC4      -0.149189096 -0.044283111 -0.084404694
#> IC5      0.603076352  -0.030319023 -0.001772235
#> IC6      0.005530295  0.101156377 -0.017695373

```

These correlation matrix can be visualized in many ways. We propose a *radar plot* to evaluate quickly if there is a good match between metagenes and components. In this representation, we focus on positive correlations that have red color and placed away from the center of the circle (radar). We can see which component is attributed to the highest correlation for each metagene.

```
p <- radar_plot_corr(corr, size.el.txt = 10, point.size = 2)
```



The function `radar_plot_corr()` returns as well the matrix in long format suitable for `ggplot2` plots in case you want to use a different type of visualization.

```

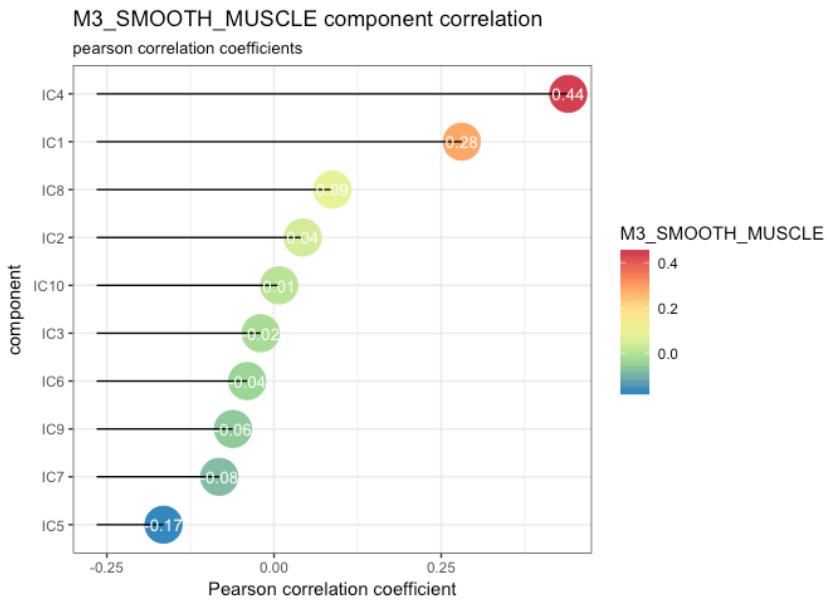
head(p$df)
#>   component  correlation       metagene
#> 1    IC1  5.241598e-03 M12_MYOFIBROBLASTS
#> 2    IC2  3.872262e-02 M12_MYOFIBROBLASTS
#> 3    IC3  8.789440e-02 M12_MYOFIBROBLASTS
#> 4    IC4  7.534325e-01 M12_MYOFIBROBLASTS
#> 5    IC5 -1.393165e-05 M12_MYOFIBROBLASTS
#> 6    IC6  1.442049e-02 M12_MYOFIBROBLASTS

```

In order to *zoom in* into a correlation with a specific metagene, one can use a function `lolypop_plot_corr()`

Here we can visualize for example SMOOTH_MUSCLE metagene that seems a bit ambiguous.

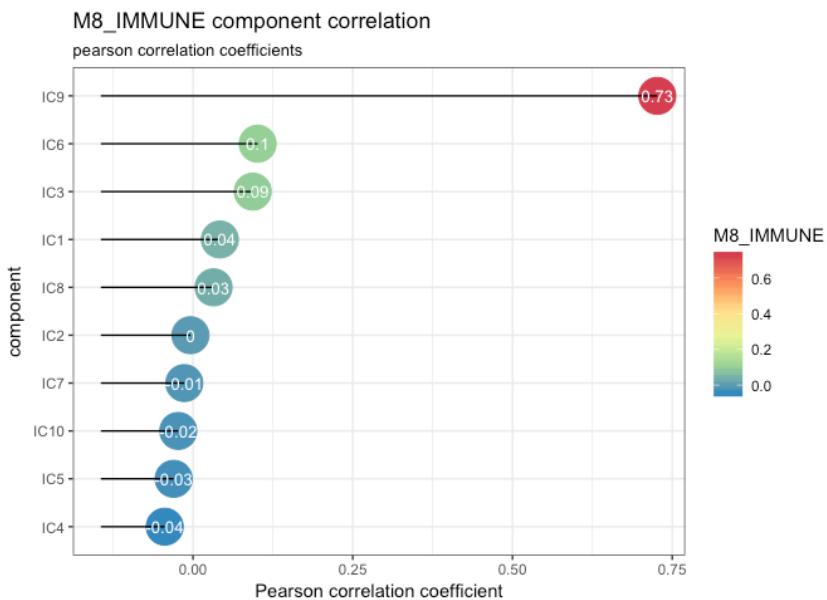
```
lolypop_plot_corr(corr$r,"M3_SMOOTH_MUSCLE")
```



We can observe that the highest correlation is with IC4 0.44 but IC1: 0.28 is close as well. These two components may represent two different types of functions or cells related to muscles.

However, if we look at IMMUNE metagene, we can select one strongly correlated component: IC9, Pearson correlation coefficient equal to 0.73.

```
lolypop_plot_corr(corr$r,"M8_IMMUNE")
```



In case we have many profiles, an automatic extraction of corresponding pairs metagene - component is handy.

A natural way is consider the component which Pearson correlation coefficient is highest as the one corresponding to a metagene. You can use `get_max_correlations()` to retrieve this information from the correlation matrix.

```
# retrieve max correlations
max.corr <- get_max_correlations(corr)
# order
max.corr.ordered <- max.corr[order(-max.corr$r),]
# show table
kable(max.corr.ordered,"html", row.names = FALSE)
```

TYPE	IC	r	p.val
M2_GC_CONTENT	IC10	0.7812173	0.0000000
M12_MYOFIBROBLASTS	IC4	0.7534325	0.0000000
M8_IMMUNE	IC9	0.7266403	0.0000000
M7_CELLCYCLE	IC5	0.6030764	0.0000000
M3_SMOOTH_MUSCLE	IC4	0.4400201	0.0000000
M5_INTERFERON	IC9	0.4092376	0.0000000
M4_MITOCHONRIA_TRANSLATION	IC6	0.3450359	0.0000000
M6_BASALLIKE	IC8	0.2297117	0.0000000
M14_STRESS	IC4	0.2218676	0.0000000
M13_BLCAPATHWAYS	IC1	0.1155118	0.0000000
M9_UROTHELIALDIFF	IC3	0.0370318	0.0002561

`get_max_correlations()` provides pearson correlation `r` column and the p-value `p.val` to help decide if the maximal correlation can be used as labelling. One can decide on minimal threshold, or p-value to take a decision.

Another way to assign metagene to a component can be trough reciprocal correlations. This method was used in our research articles [Becht et al. (2016); RBH_paper]. In brief, given correlations between the set of metagenes $M = \{M_1, \dots, M_m\}$ and S matrix $S = \{IC_1, \dots, IC_N\}$, if $S_i = argmax_k(corr(M_j, S_k))$ and $A_j = argmax_k(corr(S_i, M_k))$, then S_i and M_j are reciprocal. This approach is useful with assumption that there should be one component corresponding to a metagene and one metagene corresponding to a component.

```
# retrieve reciprocal correlations
reciprocal.corr <- assign_metagenes(corr$r, exclude_name = NULL)
#> no profiles to exclude provided
#> DONE
# show table
kable(reciprocal.corr, "html", row.names = FALSE)
```

profile	component
M12_MYOFIBROBLASTS	IC4
M2_GC_CONTENT	IC10
M4_MITOCHONRIA_TRANSLATION	IC6
M6_BASALLIKE	IC8
M7_CELLCYCLE	IC5
M8_IMMUNE	IC9

Here the corresponding pairs do not follow a specific order. The six of components find a reciprocal match.

Full pipeline example

Decompose data

In order to fully explore a dataset with identification and quantification of immune-related signals we suggest to over decompose the data matrix.

We recommend to use for this purpose MATLAB implementation of the algorithm (see [vignette: Running fastICA with icasso stabilisation](#))

One can run it like this on BEK complete data (WARNING: requires MATLAB and take a few minutes):

```
library(deconica)
BEK_ica_overdecompose <- run_fastica (
  BEK,
  isLog = FALSE,
  overdecompose = TRUE,
  with.names = FALSE,
  gene.names = row.names(BEK),
  R = FALSE
)
```

In order to follow this tutorial simply load precomputed ICA decomposition.

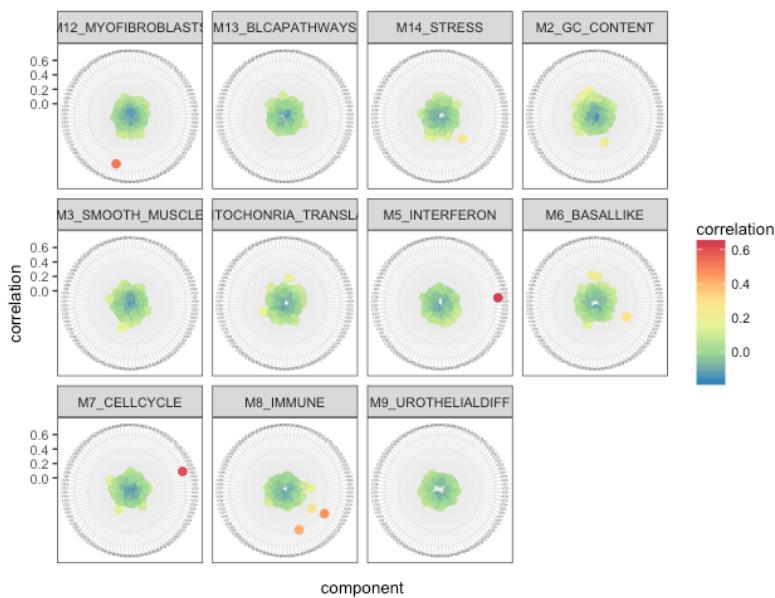
```
data(BEK_ica_overdecompose)
```

Interpret components

```
# correlate with Biton et al. metagenes
corr_Biton <- correlate_metagenes(S = BEK_ica_overdecompose$S,
                                    gene.names = BEK_ica_overdecompose$names,
                                    metagenes = Biton.list
)
# correlate with LM22 cell profiles
corr_LM22 <- correlate_metagenes(S = BEK_ica_overdecompose$S,
                                    gene.names = BEK_ica_overdecompose$names,
                                    metagenes = LM22.list
)
```

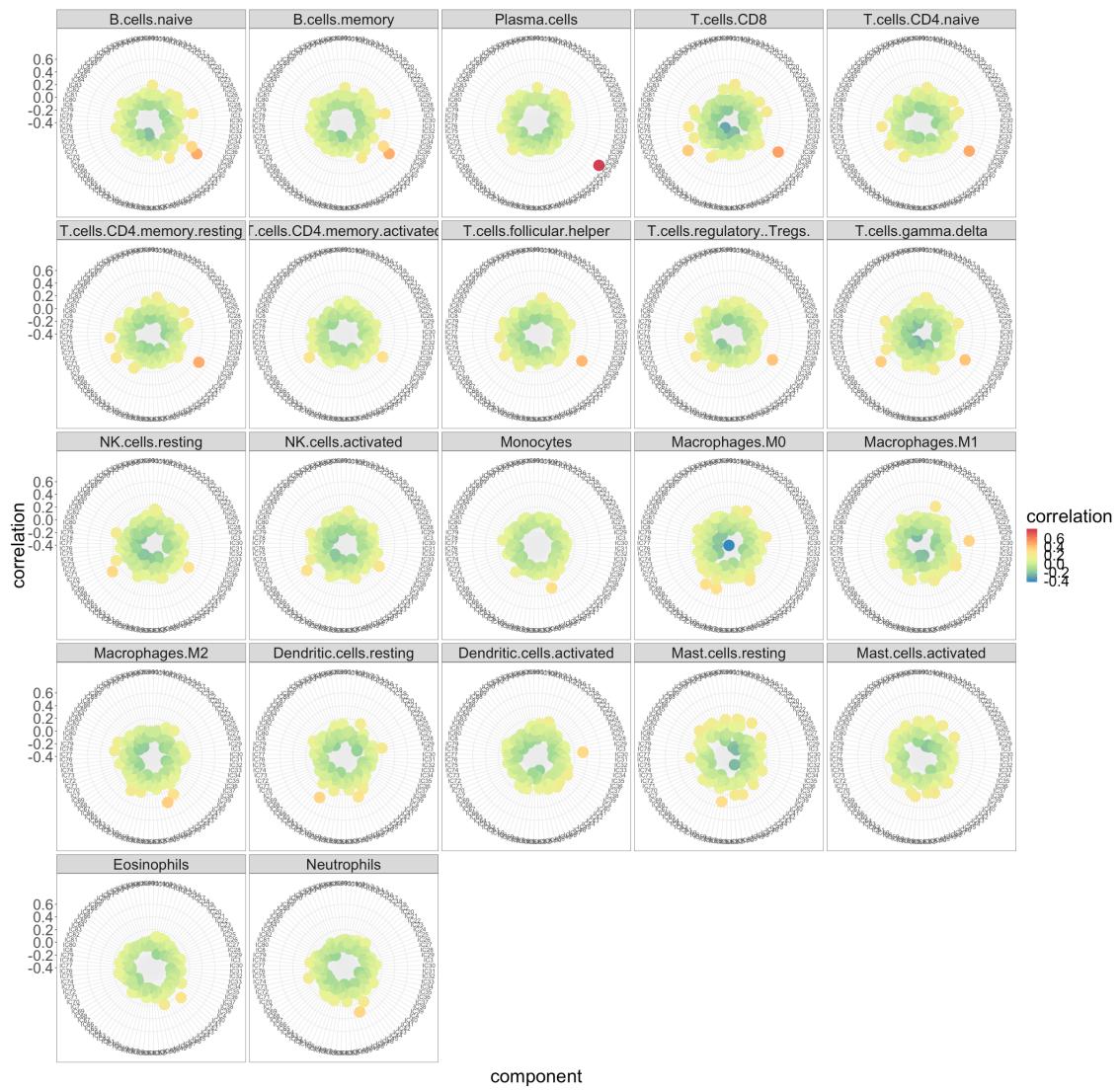
We can illustrate correlations with reference metagenes...

```
radar_plot_corr(corr_Biton, size.el.txt = 10, point.size = 2)
```



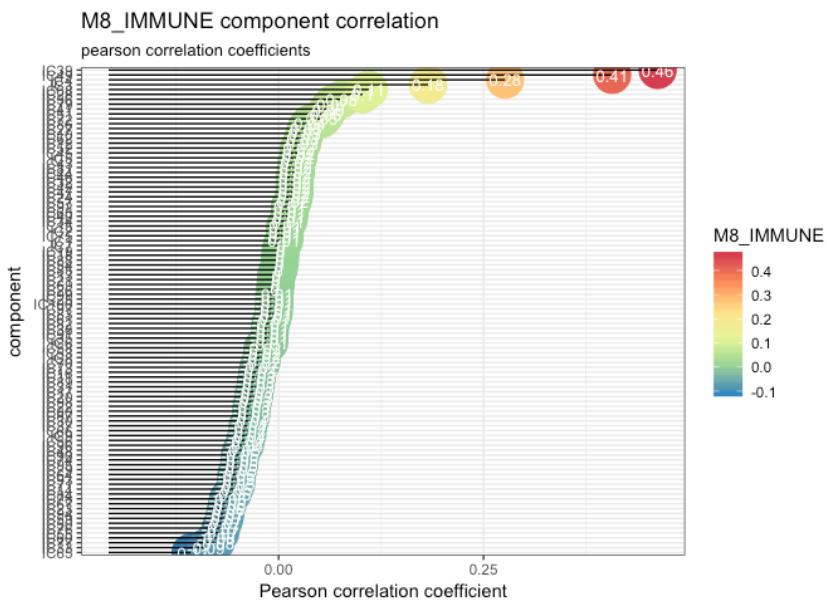
... and with LM22 immune cell type profiles.

```
radar_plot_corr(corr_LM22, size.el.txt = 10, point.size = 2)
```



And zoom in the *M8_IMMUNE* metagene.

```
lolypop_plot_corr(corr_Biton$r, "M8_IMMUNE")
```



As we have several components corresponding to the M8_IMMUNE and a few matches with immune profiles. One can use enrichment test to confirm this results.

First we run reciprocal assignment so that we can exclude from immune signals the confounders as, for example, cell cycle.

```
reciprocal.corr.Biton <-  
  assign_metagenes(corr_Biton$r, exclude_name = c("M8_IMMUNE", "M2_GC_CONTENT"))  
#> profiles excluded  
#> DONE  
immune.components <-  
  identify_immune_comp(corr_Biton$r[, "M8_IMMUNE"], reciprocal.corr.Biton$component)
```

Five components pass the threshold of >0.1 pearson correlation with the IMMUNE component.

```
immune.components
#>      IC4      IC28      IC39      IC49      IC68
#> 0.2763554 0.1023885 0.4624926 0.4070063 0.1100752
```

We can verify to which cells they may correspond through enrichment test (based on Fisher exact test).

```
enrichment.immune <- gene_enrichment_test(
  BEK_ica_overdecompose$S,
  BEK_ica_overdecompose$names,
  immune.ics = names(immune.components),
  alternative = "greater",
  p.adjust.method = "BH",
  p.value.threshold = 0.05
)
#> 58 modules higher than threshold: 500
#> Warning in gene_enrichment_test(BEK_ica_overdecompose$S,
#> BEK_ica_overdecompose$names, : Small overlap between provided gene list and
#> gmt signatures: 5450/21320
#> saving metagenes
#> extracting top genes
#>
#> running enrichment for: IC4
#> correcting p.values
#> applying p.value.threshold
#> running enrichment for: IC28
#> correcting p.values
#> applying p.value.threshold
#> running enrichment for: IC39
```

```

#> correcting p.values
#> applying p.value.threshold
#> running enrichment for: IC49
#> correcting p.values
#> applying p.value.threshold
#> running enrichment for: IC68
#> correcting p.values
#> applying p.value.threshold
#>
#> DONE

```

```

names(enrichment.immune)
#> [1] "metagenes" "genes.List" "enrichment"

```

Output of `gene_enrichment_test()` is a list of three different elements for each component passing fixed p-value threshold. `metagenes` is a weighted list of top n genes, `genes.list` is character vector of top n genes and `enrichment` is enrichment test result

```

enrichment.immune$metagenes$IC4[1:10,]
#>      gene.names    IC4
#> IGLV1-44   IGLV1-44 29.53511
#> JCHAIN      JCHAIN 23.49211
#> IGKV1D-13  IGKV1D-13 22.57536
#> IGLJ3       IGLJ3 22.20133
#> IGHM        IGHM 20.92393
#> POU2AF1     POU2AF1 20.50039
#> IGLV@       IGLV@ 20.45265
#> IGLC1       IGLC1 20.42825
#> IGH         IGH 20.20349
#> IGHD        IGHD 19.02722

```

```

enrichment.immune$genes.list$IC4[1:10]
#> [1] "JCHAIN"   "POU2AF1"  "FAM46C"  "MZB1"    "SLAMF7"   "CD38"    "IRF4"
#> [8] "CD79A"    "SEL1L3"   "CXCL9"

```

```

kable(enrichment.immune$enrichment$IC4[1:3,], "html", row.names = FALSE) %>%
  kable_styling(font_size = 8)

```

module	module_size	nb_genes_in_module	genes_in_module	universe_size	nb_genes_in_universe	p.value	p.value.corrected	test
gamma.delta.T.cells	926	22	JCHAIN POU2AF1 FAM46C MZB1 SLAMF7 CD38 IRF4 CD79A SEL1L3 CXCL9 AMPD1 CD27 PDK1 RBP1 LAX1 PIM2 CH3L1 CYTIP RASSF6 UBD LTF CCL5	5723	5450	5.52377910001422e-17		0 greater
alpha.beta.T.cells	432	21	JCHAIN POU2AF1 FAM46C MZB1 SLAMF7 CD38 IRF4 CD79A SEL1L3 CXCL9 AMPD1 CD27 PDK1 RBP1 LAX1 PIM2 CH3L1 CYTIP RASSF6 UBD LTF	5723	5450	3.34922808722385e-16		0 greater
Myeloid.Cells	952	21	JCHAIN POU2AF1 FAM46C MZB1 SLAMF7 CD38 IRF4 CD79A SEL1L3 CXCL9 AMPD1 CD27 PDK1 RBP1 LAX1 PIM2 CH3L1 CYTIP RASSF6 UBD LTF	5723	5450	3.34922808722385e-16		0 greater

If you use ImmgenHUGO list of signatures, `cell_voting_immgene()` can be used to summarize results.

```

kable(
  cell_voting_immgene(enrichment.immune$enrichment)$IC4,
  "html",
  row.names = FALSE,
  caption = "IC4"
)

```

IC4

cell.type	vote
NK.cells	68.21 %
gamma.delta.T.cells	20.84 %
alpha.beta.T.cells	8.27 %
Myeloid.Cells	2.68 %

```
kable(
  cell_voting_immgene(enrichment.immune$enrichment)$IC28,
  "html",
  row.names = FALSE,
  caption = paste("IC28")
)
```

IC28

cell.type	vote
Stromal.Cells	75.22 %
B.cells	15.91 %
Myeloid.Cells	8.87 %

```
kable(
  cell_voting_immgene(enrichment.immune$enrichment)$IC39,
  "html",
  row.names = FALSE,
  caption = paste("IC39")
)
```

IC39

cell.type	vote
gamma.delta.T.cells	71.58 %
alpha.beta.T.cells	28.42 %

```
kable(
  cell_voting_immgene(enrichment.immune$enrichment)$IC49,
  "html",
  row.names = FALSE,
  caption = paste("IC49")
)
```

IC49

cell.type	vote
Myeloid.Cells	100 %

```
kable(
  cell_voting_immgene(enrichment.immune$enrichment)$IC68,
  "html",
  row.names = FALSE,
  caption = paste("IC68")
)
```

IC68

cell.type	vote
-----------	------

cell.type	vote
gamma.delta.T.cells	57.6 %
Stromal.Cells	23.56 %
alpha.beta.T.cells	11.43 %
Myeloid.Cells	7.41 %

This result is not trivial to interpret.

A different .gmt can be also used to perform the enrichment analysis.

We can import for example signature genes from *TIMER* (Li et al. 2016)

```
setwd(path.package("deconica", quiet = TRUE))
TIMER <-
  ACSNMineR::format_from_gmt("./data-raw/TIMER_cellTypes.gmt")
```

```
enrichment.immune <- gene_enrichment_test(
  BEK_ica_overdecompose$/,
  BEK_ica_overdecompose$names,
  immune.ics = names(immune.components),
  gmt = TIMER,
  alternative = "greater",
  p.adjust.method = "BH",
  p.value.threshold = 0.05
)
#> 1 modules higher than threshold: 500
#> Warning in gene_enrichment_test(BEK_ica_overdecompose$/,
#> BEK_ica_overdecompose$names, : Small overlap between provided gene list and
#> gmt signatures: 673/21320
#> saving metagenes
#> extracting top genes
#>
#> running enrichment for: IC4
#> correcting p.values
#> applying p.value.threshold
#> running enrichment for: IC28
#> correcting p.values
#> applying p.value.threshold
#> running enrichment for: IC39
#> correcting p.values
#> applying p.value.threshold
#> running enrichment for: IC49
#> correcting p.values
#> applying p.value.threshold
#> running enrichment for: IC68
#> correcting p.values
#> applying p.value.threshold
#>
#> DONE
```

```
kable(enrichment.immune$enrichment$IC4, "html", row.names = FALSE, caption = "IC4") %>%
  kable_styling(font_size = 8)
```

IC4								
module	module.size	nb_genes_in_module	genes_in_module	universe_size	nb_genes_in_universe	p.value	p.value.corrected	test
Lymphoid	234	41	POU2AF1 TNFRSF17 FCRL5 CD79A AMPD1 CD27 PDK1 PIM2 CHI3L1 CPNE5 PNOC FKBP11 IDO1 CD2 CXCL11 CEP128 ELL2 EA2 CDBA MANEA ITK MMP12 LCK CCND2 CYP1B1 NLRC3 TPPI2 GZMK CD274 UBE2J1 GZMA MS4A1 CD19 KLF12 CCL19 EOMES AIM2 FAM30A HSPA13 NKX7 SDF2L1	924	673	7.9891782953872e-33	0.000000	greater

module	module_size	nb_genes_in_module	genes_in_module	universe_size	nb_genes_in_universe	p.value	p.value.corrected	test
B Cell	91	22	POU2AF1 TNRSF17 FCRL5 CD79A AMPD1 CD27 PDK1 PIM2 CHI3L1 CPNE5 PNOC FKBP11 IDO1 CD2 CXCL11 CEP128 ELL2 EAFC2 C9BA MANEA ITK MMP12	924	673	5.52377910001422e-17	0.000000	greater
Myeloid	344	18	POU2AF1 TNRSF17 FCRL5 CD79A AMPD1 CD27 PDK1 PIM2 CHI3L1 CPNE5 PNOC FKBP11 IDO1 CD2 CXCL11 CEP128 ELL2 EAFC2	924	673	7.0143784551178e-14	0.000000	greater
T Cell	76	10	POU2AF1 TNRSF17 FCRL5 CD79A AMPD1 CD27 PDK1 PIM2 CHI3L1 CPNE5	924	673	7.04213335186756e-08	0.000002	greater
Multiple	795	9	POU2AF1 TNRSF17 FCRL5 CD79A AMPD1 CD27 PDK1 PIM2 CHI3L1	924	673	3.79965656677689e-07	0.000007	greater
Dendritic Cell	70	5	POU2AF1 TNRSF17 FCRL5 CD79A AMPD1	924	673	0.000294962142828159	0.0003792	greater
Neutrophil	45	5	POU2AF1 TNRSF17 FCRL5 CD79A AMPD1	924	673	0.000294962142828159	0.0003792	greater
Monocyte	82	3	POU2AF1 TNRSF17 FCRL5	924	673	0.0078038786005748	0.0087844	greater

```
kable(enrichment.immune$enrichment$IC28, "html", row.names = FALSE, caption = "IC28") %>%  
kable_styling(font_size = 8)
```

IC28								
module	module_size	nb_genes_in_module	genes_in_module	universe_size	nb_genes_in_universe	p.value	p.value.corrected	test
Myeloid	344	31	MMP12 MS4A1 SPIB MMP9 CXCL5 SGPP2 BCL11A KRT23 P2RX5 CHI3L1 CYP1B1 WNT5A L32 CD19 RASSF4 POU2AF1 NRIP3 CD1A BCL2A1 AC5 FCMR SERPINB2 CCL19 IRAK2 GPR18 FCRL5 KYNU ARL4C STAP1 TRIM59 NLRP7	924	673	3.02686371304923e-24	0.000000	greater
Lymphoid	234	24	MMP12 MS4A1 SPIB MMP9 CXCL5 SGPP2 BCL11A KRT23 P2RX5 CHI3L1 CYP1B1 WNT5A L32 CD19 RASSF4 POU2AF1 NRIP3 CD1A BCL2A1 AC5 FCMR SERPINB2 CCL19 IRAK2	924	673	1.45504141446656e-18	0.000000	greater
B Cell	91	20	MMP12 MS4A1 SPIB MMP9 CXCL5 SGPP2 BCL11A KRT23 P2RX5 CHI3L1 CYP1B1 WNT5A L32 CD19 RASSF4 POU2AF1 NRIP3 CD1A BCL2A1 AC5	924	673	2.00953685233431e-15	0.000000	greater
Monocyte	82	11	MMP12 MS4A1 SPIB MMP9 CXCL5 SGPP2 BCL11A KRT23 P2RX5 CHI3L1 CYP1B1	924	673	1.29345306462873e-08	0.000000	greater
Dendritic Cell	70	10	MMP12 MS4A1 SPIB MMP9 CXCL5 SGPP2 BCL11A KRT23 P2RX5 CHI3L1	924	673	7.04213335186756e-08	0.000001	greater
Multiple	795	10	MMP12 MS4A1 SPIB MMP9 CXCL5 SGPP2 BCL11A KRT23 P2RX5 CHI3L1	924	673	7.04213335186756e-08	0.000001	greater
T Cell	76	5	MMP12 MS4A1 SPIB MMP9 CXCL5	924	673	0.000294962142828159	0.0003792	greater

```
kable(enrichment.immune$enrichment$IC39, "html", row.names = FALSE, caption = "IC39") %>%  
kable_styling(font_size = 8)
```

IC39								
module	module_size	nb_genes_in_module	genes_in_module	universe_size	nb_genes_in_universe	p.value	p.value.corrected	test
Lymphoid	234	50	MS4A1 CCL19 FCRL3 MMP9 GPR18 EOMES NLRC3 GZMK BCL11B ITK LCK FCMR CD2 TCL1A TMGB CD19 Q247 CD3E IL2RB CD3D GZMA CD27 P2RX5 CD69 POU2AF1 PTX3 CD8A GZMB RASGRP2 SPIB SAMD3 STAT4 MAP4K1 BANK1 PTPN7 ZAP70 TRAT1 TLR10 PTRPCAP FCRL1 CD79A KRT23 GNLY NKKG7 MYBL1 KLRL1 RASSF2 BCL11A KLRL1 CD6	924	673	4.35924427142961e-41	0.000000	greater
T Cell	76	15	MS4A1 CCL19 FCRL3 MMP9 GPR18 EOMES NLRC3 GZMK BCL11B ITK LCK FCMR CD2 TCL1A TMGB	924	673	1.34198340448245e-11	0.000000	greater
B Cell	91	13	MS4A1 CCL19 FCRL3 MMP9 GPR18 EOMES NLRC3 GZMK BCL11B ITK LCK FCMR CD2	924	673	4.24516594680188e-10	0.000000	greater

module	module_size	nb_genes_in_module	genes_in_module	universe_size	nb_genes_in_universe	p.value	p.value.corrected	test
Myeloid	344	13	MS4A1 CCL19 FCRL3 MMP9 GPR18 EOMES NLRC3 GZMK BCL11B ITK LCK FCMLR CD2	924	673	4.24516594680188e-10	0.000000	greater
Multiple	795	8	MS4A1 CCL19 FCRL3 MMP9 GPR18 EOMES NLRC3 GZMK	924	673	2.03199025092851e-06	0.0000037	greater
Monocyte	82	5	MS4A1 CCL19 FCRL3 MMP9 GPR18	924	673	0.000294962142828159	0.0004424	greater
Dendritic Cell	70	4	MS4A1 CCL19 FCRL3 MMP9	924	673	0.00152397107127882	0.0019594	greater

```
kable(enrichment.immune$enrichment$IC49, "html", row.names = FALSE, caption = "IC49") %>%
kable_styling(font_size = 8)
```

IC49								
module	module_size	nb_genes_in_module	genes_in_module	universe_size	nb_genes_in_universe	p.value	p.value.corrected	test
Myeloid	344	69	FCGR3B FGL2 FCGR2B TLR8 CPVL MPEG1 CD86 CYBB TLR1 MNDA MS4A6A PRF3 CSF1R P2RX7 GGTAT1P TEC CXCL11 HLA-DRB4 NCFL2 P2RY13 HCK TLR2 CSAR1 MS4A7 THEMIS2 CSF2RA CLECSA CLEC7A KCTD12 RASSF4 IGSF6 EMILIN2 AIF1 SLAMF8 PILRA TREM2 DSE NCFL4 DPYD GZMA FPR1 CPM NPL LLRB2 TNFSF13B ARHGAP18 ADAP2 STAR1 HPSE PTAFR LYN LLRB1 AP1S2 HSP60 HMOX1 SIGLEC1 SLC15A3 CD8A KYNU DFNA5 TNFAIP2 MAN1A1 MMP1 CLIC2 CD69 XCL1 IL15 SOD2 MAFB	924	673	9.19957214014236e-61	0.000000	greater
Lymphoid	234	10	FCGR3B FGL2 FCGR2B TLR8 CPVL MPEG1 CD86 CYBB TLR1 MNDA	924	673	7.04213335186756e-08	0.000002	greater
Monocyte	82	10	FCGR3B FGL2 FCGR2B TLR8 CPVL MPEG1 CD86 CYBB TLR1 MNDA	924	673	7.04213335186756e-08	0.000002	greater
Dendritic Cell	70	9	FCGR3B FGL2 FCGR2B TLR8 CPVL MPEG1 CD86 CYBB TLR1	924	673	3.79965656677689e-07	0.000009	greater
Multiple	795	6	FCGR3B FGL2 FCGR2B TLR8 CPVL MPEG1	924	673	5.66088960983336e-05	0.0001019	greater
Neutrophil	45	4	FCGR3B FGL2 FCGR2B TLR8	924	673	0.00152397107127882	0.0022860	greater

```
kable(enrichment.immune$enrichment$IC68, "html", row.names = FALSE, caption = "IC68") %>%
kable_styling(font_size = 8)
```

IC68								
module	module_size	nb_genes_in_module	genes_in_module	universe_size	nb_genes_in_universe	p.value	p.value.corrected	test
Lymphoid	234	42	PTGDR GZMB GZMK FCMR IL32 CD8A XCL2 GZMA NK7 VCPKMT RIT1 IL2RB BCL11A ZBED2 XCL1 INHBA CD2 FCRL3 HSPA13 TNFRSF18 EOMES GNLY DUSP6 MS4A1 PLA2G7 TYMP ATP2B1 CD3D MANEA CAPG SAMD3 BTN3A2 LAG3 NLRC3 PRF1 OSBP11 AIFM2 CCL19 SPRED2 ACER3 ZAP70 SPIB	924	673	1.0269314148755e-33	0.000000	greater
Myeloid	344	25	PTGDR GZMB GZMK FCMR IL32 CD8A XCL2 GZMA NK7 VCPKMT RIT1 IL2RB BCL11A ZBED2 XCL1 INHBA CD2 FCRL3 HSPA13 TNFRSF18 EOMES GNLY DUSP6 MS4A1 PLA2G7	924	673	2.32317536763567e-19	0.000000	greater
B Cell	91	13	PTGDR GZMB GZMK FCMR IL32 CD8A XCL2 GZMA NK7 VCPKMT RIT1 IL2RB BCL11A	924	673	4.24516594680188e-10	0.000000	greater
T Cell	76	13	PTGDR GZMB GZMK FCMR IL32 CD8A XCL2 GZMA NK7 VCPKMT RIT1 IL2RB BCL11A	924	673	4.24516594680188e-10	0.000000	greater
Dendritic Cell	70	9	PTGDR GZMB GZMK FCMR IL32 CD8A XCL2 GZMA NK7	924	673	3.79965656677689e-07	0.000006	greater
Multiple	795	9	PTGDR GZMB GZMK FCMR IL32 CD8A XCL2 GZMA NK7	924	673	3.79965656677689e-07	0.000006	greater
Neutrophil	45	3	PTGDR GZMB GZMK	924	673	0.00078038786005748	0.0100394	greater
Monocyte	82	2	PTGDR GZMB	924	673	0.0396793587174349	0.0396794	greater
NK Cell	16	2	PTGDR GZMB	924	673	0.0396793587174349	0.0396794	greater

Based on this results we can suppose that:

- IC4 is related to B-cells
- IC49 and IC68 are related to Myeloid cells
- IC39 is related to T-cells
- IC28 can be characterized as stroma

This result can be cross-verified with radar plot and maximal correlations with LM22

```
#retrieve max correlations
max.corr <- get_max_correlations(corr_LM22)
# order
max.corr.ordered <- max.corr[order(-max.corr$r),]
# show table
kable(max.corr.ordered, "html", row.names = FALSE)
```

TYPE	IC	r	p.val
Plasma.cells	IC4	0.7880667	0e+00
T.cells.CD8	IC39	0.4987960	0e+00
T.cells.CD4.naive	IC39	0.4705819	0e+00
T.cells.CD4.memory.resting	IC39	0.4691812	0e+00
B.cells.naive	IC4	0.4644702	0e+00
B.cells.memory	IC4	0.4598657	0e+00
T.cells.follicular.helper	IC39	0.4300112	0e+00
T.cells.gamma.delta	IC39	0.3949245	0e+00
T.cells.regulatory..Tregs.	IC39	0.3940503	0e+00
Macrophages.M2	IC49	0.3654883	0e+00
Macrophages.M1	IC3	0.3499043	0e+00
Dendritic.cells.resting	IC61	0.3447491	0e+00
Neutrophils	IC49	0.3371526	0e+00
NK.cells.resting	IC68	0.3338840	0e+00
Dendritic.cells.activated	IC3	0.3332215	0e+00
Mast.cells.resting	IC56	0.3168765	0e+00
Monocytes	IC49	0.3099286	0e+00
Macrophages.M0	IC61	0.3073292	0e+00
T.cells.CD4.memory.activated	IC39	0.3059700	0e+00
NK.cells.activated	IC68	0.2991302	0e+00
Eosinophils	IC42	0.2613817	0e+00
Mast.cells.activated	IC56	0.2329371	1e-07

Indeed from correlation with LM22 we learn that

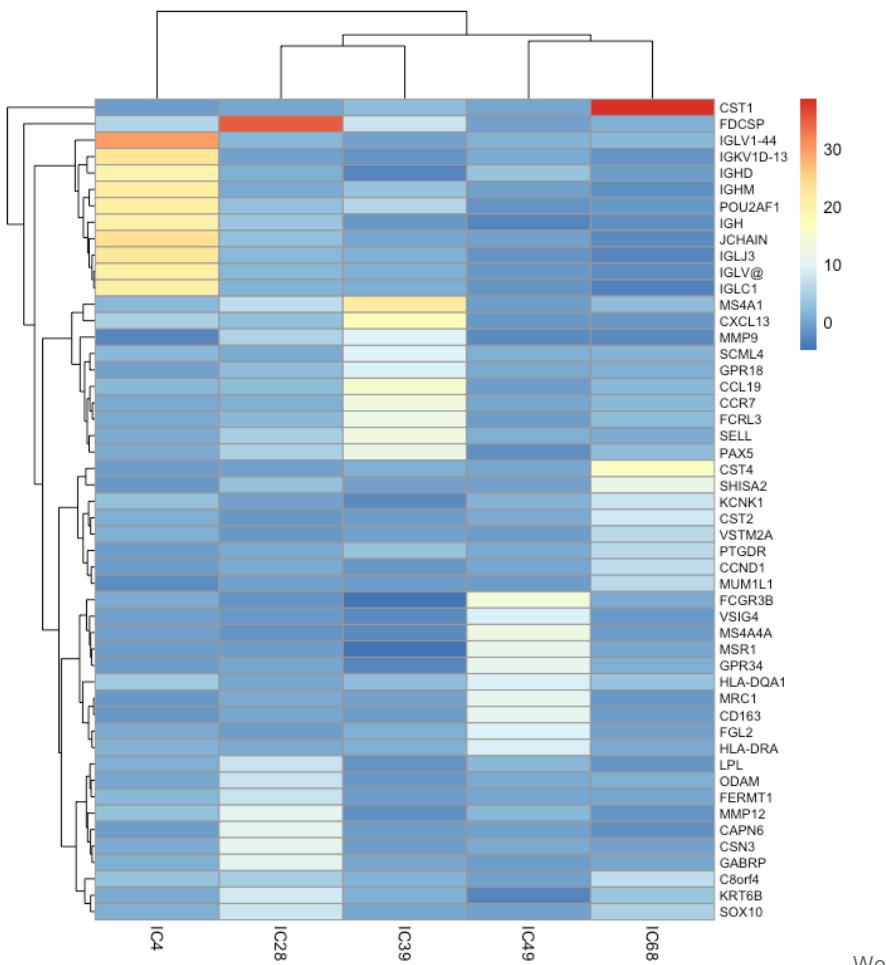
- IC4 is related to B-cells
- IC39 to T-cells
- IC49 to M2 Macrophages, Monocytes and Neutrophils
- IC68 to NK

Estimate abundance

Once we identified components related to the immune cells, we can use them to extract *specific* markers, best number of markers usually vary from 10 to 30

```
markers_10 <-  
  generate_markers(  
    df = BEK_ica_overdecompose,  
    n = 10,  
    sel.comp = names(immune.components),  
    return = "gene.list"  
)  
  
#> markers_10  
#> $IC4  
#> [1] "IGLV1-44"   "JCHAIN"      "IGKV1D-13"  "IGLJ3"      "IGHM"  
#> [6] "POU2AF1"   "IGLV@"       "IGLC1"       "IGH"        "IGHD"  
#>  
#> $IC28  
#> [1] "FDCSP"     "CSN3"       "GABRP"      "MMP12"      "CAPN6"      "KRT6B"      "SOX10"  
#> [8] "LPL"        "ODAM"       "FERMT1"  
#>  
#> $IC39  
#> [1] "MS4A1"     "CXCL13"     "CCL19"      "CCR7"       "SELL"       "FCRL3"      "PAX5"  
#> [8] "SCML4"     "MMP9"       "GPR18"  
#>  
#> $IC49  
#> [1] "FCGR3B"    "MS4A4A"     "MSR1"       "GPR34"      "MRC1"       "CD163"  
#> [7] "FGL2"       "HLA-DQA1"   "VSIG4"      "HLA-DRA"  
#>  
#> $IC68  
#> [1] "CST1"       "CST4"       "SHISA2"     "CST2"       "KCNK1"      "CCND1"      "C8orf4"  
#> [8] "MUM1L1"    "PTGDR"     "VSTM2A"
```

```
basis_10 <-  
  generate_basis(  
    df = BEK_ica_overdecompose,  
    sel.comp = names(immune.components),  
    markers = markers_10  
)  
pheatmap::pheatmap(basis_10, fontsize_row = 8)
```

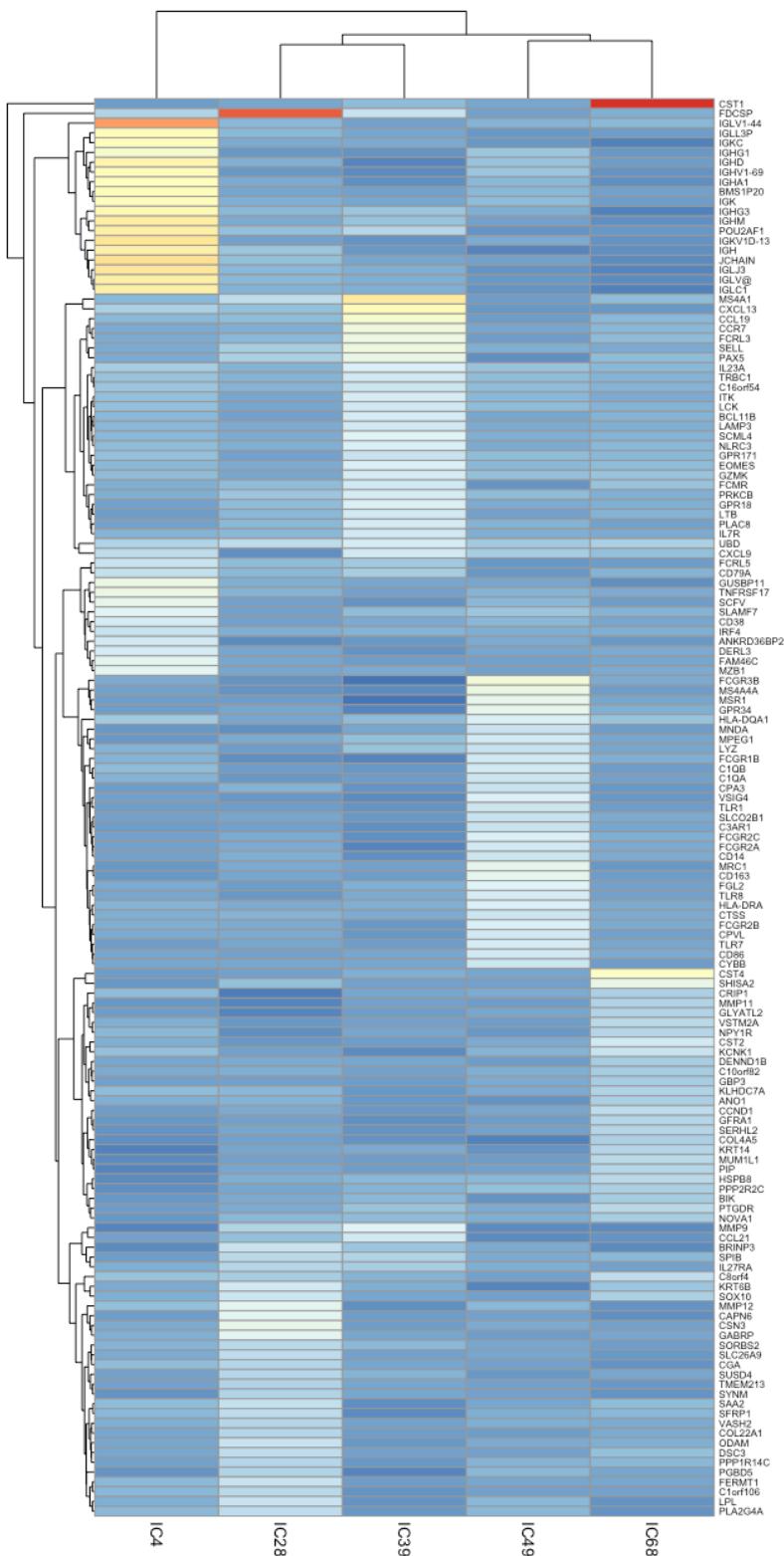


We can observe how it

looks for `n = 30` (30 specific genes for each cell type - component)

```
markers_30 <-  
  generate_markers(  
    df = BEK_ica_overdecompose,  
    n = 30,  
    sel.comp = names(immune.components),  
    return = "gene.list"  
)
```

```
basis_30 <-  
  generate_basis(  
    df = BEK_ica_overdecompose,  
    sel.comp = names(immune.components),  
    markers = markers_30  
)  
pheatmap::pheatmap(basis_30, fontsize_row = 6)
```



And finally to obtain the abundance scores, a function `get_scores()` can be used.

If you want to use only gene.list

```
head(get_scores (BEK_ica_overdecompose$counts, markers_10, summary = "mean", na.rm = TRUE))
#>          IC4      IC28      IC39      IC49      IC68
#> GSM585300 1015.6910 68.68907 142.01059 205.1942 283.7116
#> GSM585301 1309.1217 114.52619 64.02462 160.9249 316.5380
#> GSM585302 2484.4715 147.33433 561.32776 110.5064 117.8382
#> GSM585303 1641.7940 52.89996 38.92237 87.3892 208.3567
#> GSM585304 185.3009 26.68603 48.57868 128.0455 772.6966
#> GSM585305 5400.0182 98.54236 137.97633 388.9607 212.1527
```

If you want to compute weighted mean (using ICA weights), first transform to a list of data frames:

```
weighted.list_10 <- generate_markers(  
  df = BEK_ica_overdecompose,  
  n = 10,  
  sel.comp = names(immune.components),  
  return = "gene.ranked"  
)
```

```
weighted.scores <- get_scores (BEK_ica_overdecompose$counts, weighted.list_10, summary = "weighted.mean")  
head(weighted.scores)  
#> IC4      IC28      IC39      IC49      IC68  
#> GSM585300 969.9797 57.51394 139.64727 194.54404 191.63500  
#> GSM585301 1238.1501 93.36739 60.44216 152.94476 254.92725  
#> GSM585302 2386.5932 147.70302 592.17641 102.79626 95.21841  
#> GSM585303 1557.1738 41.40888 38.99046 79.37605 201.52527  
#> GSM585304 176.4919 21.41715 39.03719 123.14515 578.01167  
#> GSM585305 5195.6869 79.47580 119.84810 346.11541 138.89664
```

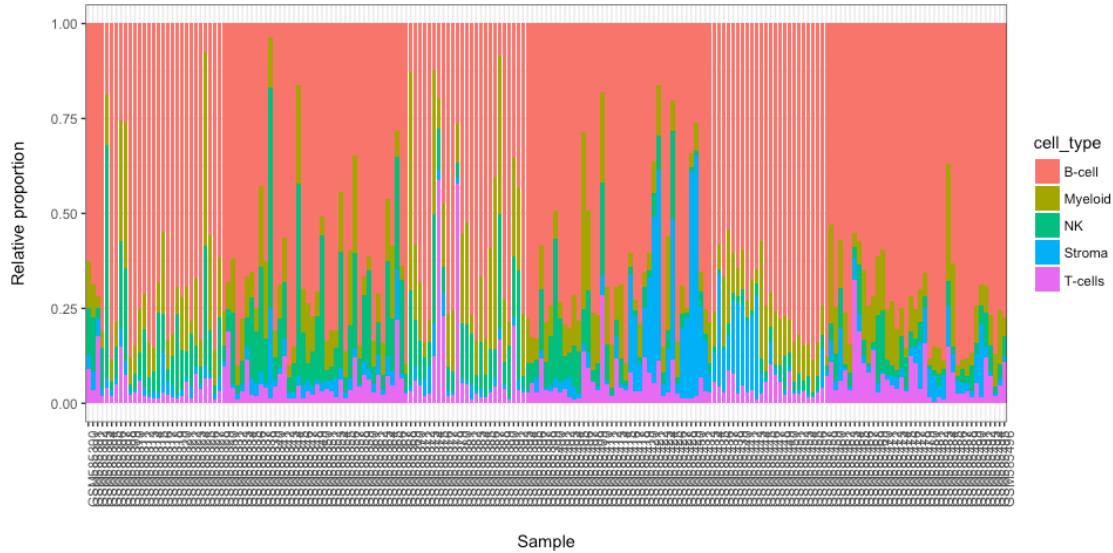
These scores are linearly correlated with proportion values of each cell type in samples.

We can rename columns according to labels we attributed

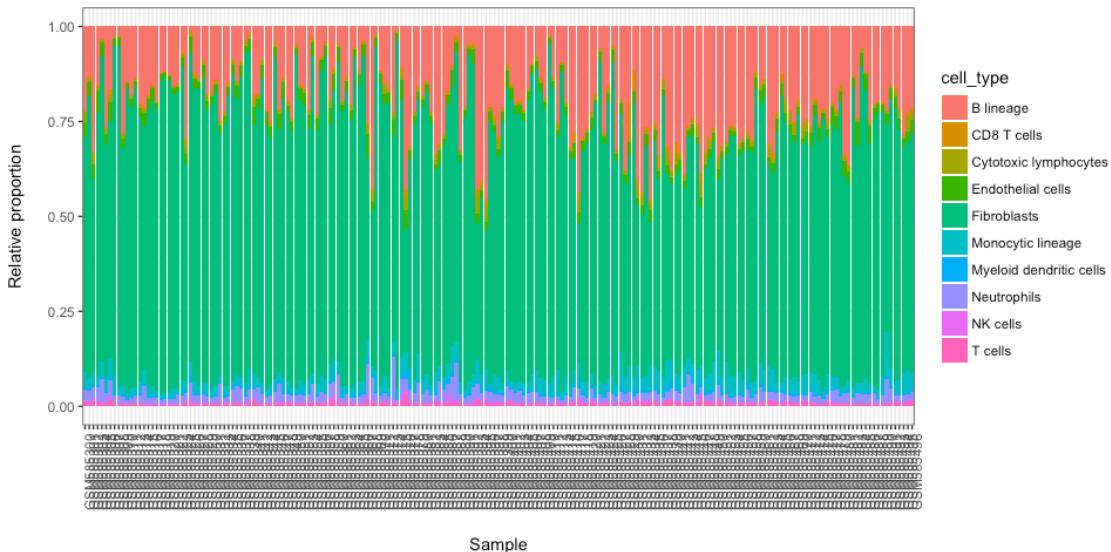
```
colnames(weighted.scores) <- c("B-cell", "Stroma", "T-cells", "Myeloid", "NK")
```

Here we present results in a form of a stacked baplot where scores sum up to 1. The scores used for plot should not be used as final scores, there are just more convinient for visualization purposes.

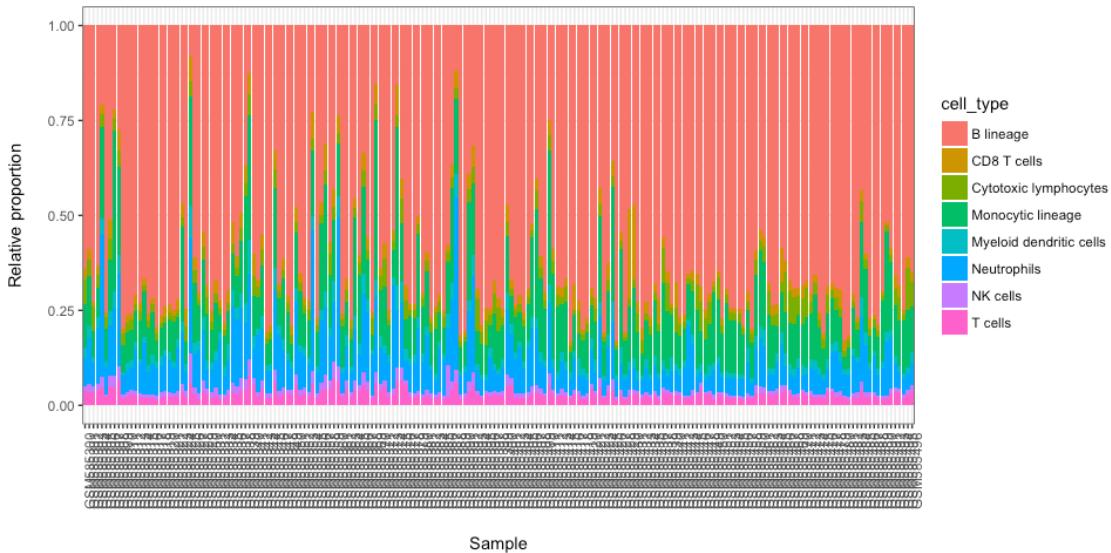
```
stacked_proportions_plot(t(weighted.scores))
```



```
library(MCPcounter)  
#> Loading required package: curl  
MCP.counter.scores <- MCPcounter.estimate(BEK_ica_overdecompose$counts, featuresType="HUGO_symbols")  
stacked_proportions_plot(MCP.counter.scores)
```



```
#without endothelial cells and Fibroblasts
stacked_proportions_plot(MCP.counter.scores[c(-10, -9),])
```



References

- Abbas, Alexander R., Kristen Wolslegel, Dhaya Seshasayee, Zora Modrusan, and Hilary F. Clark. 2009. "Deconvolution of Blood Microarray Data Identifies Cellular Activation Patterns in Systemic Lupus Erythematosus." Edited by Patrick Tan. *PLoS ONE* 4 (7). Public Library of Science: e6098. doi:[10.1371/journal.pone.0006098](https://doi.org/10.1371/journal.pone.0006098).
- Al-Ejeh, F, P T Simpson, J M Sanus, K Klein, M Kalimutho, W Shi, M Miranda, et al. 2014. "Meta-analysis of the global gene expression profile of triple-negative breast cancer identifies genes for the prognostication and treatment of aggressive breast cancer." *Oncogenesis* 3 (4). Nature Publishing Group: e100–e100. doi:[10.1038/oncsis.2014.14](https://doi.org/10.1038/oncsis.2014.14).
- Aran, Dvir, Zicheng Hu, and Atul J Butte. 2017. "xCell: digitally portraying the tissue cellular heterogeneity landscape." *Genome Biology* 18 (1). BioMed Central: 220. doi:[10.1186/s13059-017-1349-1](https://doi.org/10.1186/s13059-017-1349-1).
- Becht, Etienne, and Aurelien de Reynies. 2016. *MCPcounter: Estimating Tissue-Infiltrating Immune and Other Stromal Subpopulations Abundances Using Gene Expression*.
- Becht, Etienne, Nicolas A. Giraldo, Laetitia Lacroix, Bénédicte Buttard, Nabila Elarouci, Florent Petitprez, Janick Selvès, et al. 2016. "Estimating the population abundance of tissue-infiltrating immune and stromal cell populations using gene expression." *Genome Biology* 17 (1). doi:[10.1186/s13059-016-1070-5](https://doi.org/10.1186/s13059-016-1070-5).

- Bekhouche, Ismahane, Pascal Finetti, José Adelaïde, Anthony Ferrari, Carole Tarpin, Emmanuelle Charafe-Jauffret, Colette Charpin, et al. 2011. "High-resolution comparative genomic hybridization of Inflammatory breast cancer and identification of candidate genes." *PLoS ONE* 6 (2). doi:[10.1371/journal.pone.0016950](https://doi.org/10.1371/journal.pone.0016950).
- Biton, Anne, Isabelle Bernard-Pierrot, Yinjun Lou, Clémentine Krucker, Elodie Chapeaublanc, Carlota Rubio-Pérez, Nuria López-Bigas, et al. 2014. "Independent Component Analysis Uncovers the Landscape of the Bladder Tumor Transcriptome and Reveals Insights into Luminal and Basal Subtypes." *Cell Reports* 9 (4): 1235–45. doi:[10.1016/j.celrep.2014.10.035](https://doi.org/10.1016/j.celrep.2014.10.035).
- Brunet, Jean-Philippe, Pablo Tamayo, Todd R Golub, and Jill P Mesirov. 2004. "Metagenes and molecular pattern discovery using matrix factorization." *Proceedings of the National Academy of Sciences of the United States of America* 101 (12). National Academy of Sciences: 4164–9. doi:[10.1073/pnas.0308531101](https://doi.org/10.1073/pnas.0308531101).
- Cheng, Wei-Yi, Tai-Hsien Ou Yang, and Dimitris Anastassiou. 2013. "Biomolecular events in cancer revealed by attractor metagenes." Edited by Isidore Rigoutsos. *PLoS Computational Biology* 9 (2): e1002920. doi:[10.1371/journal.pcbi.1002920](https://doi.org/10.1371/journal.pcbi.1002920).
- Chun-Hou Zheng, De-Shuang Huang, Lei Zhang, and Xiang-Zhen Kong. 2009. "Tumor Clustering Using Nonnegative Matrix Factorization with Gene Selection." *IEEE Transactions on Information Technology in Biomedicine* 13 (4): 599–607. doi:[10.1109/TITB.2009.2018115](https://doi.org/10.1109/TITB.2009.2018115).
- Czerwinska, Urszula, Laura Cantini, Ulykbek Kairov, Emmanuel Barillot, and Andrei Zinovyev. 2018. "Application of Independent Component Analysis to Tumor Transcriptomes Reveals Specific And Reproducible Immune-related Signals." *Springer Proceedings*. LVA-ICA.
- Elmas, Abdulkadir, Tai-Hsien Ou Yang, Xiaodong Wang, and Dimitris Anastassiou. 2016. "Discovering Genome-Wide Tag SNPs Based on the Mutual Information of the Variants." Edited by Srinivas Mummid. *PLOS ONE* 11 (12). Public Library of Science: e0167994. doi:[10.1371/journal.pone.0167994](https://doi.org/10.1371/journal.pone.0167994).
- Gaujoux, Renaud, and Cathal Seoighe. 2010. "A Flexible R Package for Nonnegative Matrix Factorization." *BMC Bioinformatics* 11 (1): 367. doi:[10.1186/1471-2105-11-367](https://doi.org/10.1186/1471-2105-11-367).
- . 2012. "CellMix: A Comprehensive Toolbox for Gene Expression Deconvolution." (*Submitted*). <http://web.cbio.uct.ac.za/~renaud/CRAN/web/CellMix>.
- . 2013. *CellMix: Sample Analyses*. CRAN. <http://cran.r-project.org/package=CellMix>.
- . 2015a. *The Package Nmf: Manual Pages*. CRAN. <http://cran.r-project.org/package=NMF>.
- . 2015b. *Using the Package Nmf*. CRAN. <http://cran.r-project.org/package=NMF>.
- Gorban, A.N., Kégl, B., Wunsch, D.C., Zinovyev, A. 2008. *Principal Manifolds for Data Visualization and Dimension Reduction*. Edited by Alexander N. Gorban, Balázs Kégl, Donald C. Wunsch, and Andrei Y. Zinovyev. Vol. 58. Lecture Notes in Computational Science and Enginee. Berlin, Heidelberg: Springer Berlin Heidelberg. doi:[10.1007/978-3-540-73750-6](https://doi.org/10.1007/978-3-540-73750-6).
- Hart, Yuval, Hila Sheftel, Jean Hausser, Pablo Szekely, Noa Bossel Ben-Moshe, Yael Korem, Avichai Tendler, Avraham E Mayo, and Uri Alon. 2015. "Inferring biological tasks using Pareto analysis of high-dimensional data." *Nature Methods* 12 (3): 233–35. doi:[10.1038/nmeth.3254](https://doi.org/10.1038/nmeth.3254).
- Himberg, Johan, and Aapo Hyvärinen. 2003. "ICASSO: Software for investigating the reliability of ICA estimates by clustering and visualization." In *Neural Networks for Signal Processing - Proceedings of the Ieee Workshop*, 2003-Janua:259–68. doi:[10.1109/NNSP.2003.1318025](https://doi.org/10.1109/NNSP.2003.1318025).
- Hyvärinen, Aapo, and Erkki Oja. 2000. "Independent Component Analysis: Algorithms and Applications." *Neural Networks* 13 (45): 411–30. doi:[10.1016/S0893-6080\(00\)00026-5](https://doi.org/10.1016/S0893-6080(00)00026-5).
- Kairov, Ulykbek, Laura Cantini, Alessandro Greco, Askhat Molkenov, Urszula Czerwinska, Emmanuel Barillot, and Andrei Zinovyev. 2017. "Determining the optimal number of independent components for reproducible transcriptomic data analysis." *BMC Genomics* 18 (1). doi:[10.1186/s12864-017-4112-9](https://doi.org/10.1186/s12864-017-4112-9).
- Li, Bo, Eric Severson, Jean-Christophe Pignon, Haoquan Zhao, Taiwen Li, Jesse Novak, Peng Jiang, et al. 2016. "Comprehensive analyses of tumor immunity: implications for cancer immunotherapy." *Genome Biology* 2016 17:1 29 (1). BioMed Central: 1949–55. doi:[10.1200/JCO.2010.30.5037](https://doi.org/10.1200/JCO.2010.30.5037).

Liu, Yuan, Yu Liang, Qifan Kuang, Fanfan Xie, Yingyi Hao, Zhining Wen, and Menglong Li. 2017. "Post-modified non-negative matrix factorization for deconvoluting the gene expression profiles of specific cell types from heterogeneous clinical samples based on RNA-sequencing data." *Journal of Chemometrics*, August. Wiley-Blackwell, e2929. doi:[10.1002/cem.2929](https://doi.org/10.1002/cem.2929).

Moffitt, Richard A, Raoud Marayati, Elizabeth L Flate, Keith E Volmar, S Gabriela Herrera Loeza, Katherine A Hoadley, Naim U Rashid, et al. 2015. "Virtual microdissection identifies distinct tumor- and stroma-specific subtypes of pancreatic ductal adenocarcinoma." *Nature Genetics*. doi:[10.1038/ng.3398](https://doi.org/10.1038/ng.3398).

Newberg, Lee A., Xiaowei Chen, Chinnappa D. Kodira, and Maria I. Zavodszky. 2018. "Computational de novo discovery of distinguishing genes for biological processes and cell types in complex tissues." Edited by Paolo Provero. *PLOS ONE* 13 (3). Public Library of Science: e0193067. doi:[10.1371/journal.pone.0193067](https://doi.org/10.1371/journal.pone.0193067).

Newman, Aaron M., Chih Long Liu, Michael R. Green, Andrew J. Gentles, Weiguo Feng, Yue Xu, Chuong D. Hoang, Maximilian Diehn, and Ash A. Alizadeh. 2015. "Robust enumeration of cell subsets from tissue expression profiles." *Nature Methods* 12 (5): 453–57. doi:[10.1038/nmeth.3337](https://doi.org/10.1038/nmeth.3337).

Racle, Julien, Kaat de Jonge, Petra Baumgaertner, Daniel E Speiser, and David Gfeller. 2017. "Simultaneous enumeration of cancer and immune cell types from bulk tumor gene expression data." *eLife* 6 (November). eLife Sciences Publications, Ltd. doi:[10.7554/eLife.26476](https://doi.org/10.7554/eLife.26476).

Saidi, Samir A, Cathrine M Holland, David P Kreil, David J C MacKay, D Stephen Charnock-Jones, Cristin G Print, and Stephen K Smith. 2004. "Independent component analysis of microarray data in the study of endometrial cancer." *Oncogene* 23 (39). Nature Publishing Group: 6677–83. doi:[10.1038/sj.onc.1207562](https://doi.org/10.1038/sj.onc.1207562).

Schelker, Max, Sonia Feau, Jinyan Du, Nav Ranu, Edda Klipp, Gavin MacBeath, Birgit Schoeberl, and Andreas Raue. 2017. "Estimation of immune cell content in tumour tissue using single-cell RNA-seq data." *Nature Communications* 8 (1). Nature Publishing Group: 2032. doi:[10.1038/s41467-017-02289-3](https://doi.org/10.1038/s41467-017-02289-3).

Schwartz, Russell, and Stanley E Shackney. 2010. "Applying unmixing to gene expression data for tumor phylogeny inference." *BMC Bioinformatics* 11 (1): 42. doi:[10.1186/1471-2105-11-42](https://doi.org/10.1186/1471-2105-11-42).

Teschendorff, Andrew E, Michel Journée, Pierre A Absil, Rodolphe Sepulchre, and Carlos Caldas. 2007. "Elucidating the altered transcriptional programs in breast cancer using independent component analysis." *PLoS Computational Biology* 3 (8). Public Library of Science: e161. doi:[10.1371/journal.pcbi.0030161](https://doi.org/10.1371/journal.pcbi.0030161).

Vallania, Francesco, Andrew Tam, Shane Lofgren, Steven Schaffert, Tej D. Azad, Erika Bongen, Meia Alsup, et al. 2017. "Leveraging heterogeneity across multiple data sets increases accuracy of cell-mixture deconvolution and reduces biological and technical biases." *bioRxiv*, October. Cold Spring Harbor Laboratory, 206466. doi:[10.1101/206466](https://doi.org/10.1101/206466).

Wang, Niya, Eric P. Hoffman, Lulu Chen, Li Chen, Zhen Zhang, Chunyu Liu, Guoqiang Yu, David M. Herrington, Robert Clarke, and Yue Wang. 2016. "Mathematical modelling of transcriptional heterogeneity identifies novel markers and subpopulations in complex tissues." *Scientific Reports* 6 (1). Nature Publishing Group: 18909. doi:[10.1038/srep18909](https://doi.org/10.1038/srep18909).

Yang, Zi, and George Michailidis. 2015. "A non-negative matrix factorization method for detecting modules in heterogeneous omics multi-modal data." *Bioinformatics* 32 (1): btv544. doi:[10.1093/bioinformatics/btv544](https://doi.org/10.1093/bioinformatics/btv544).

Zinovyev, Andrei, Ulykbek Kairov, Tatyana Karpenyuk, and Erlan Ramanculov. 2013. "Blind source separation methods for deconvolution of complex signals in cancer biology." *Biochemical and Biophysical Research Communications* 430 (3): 1182–7. doi:[10.1016/j.bbrc.2012.12.043](https://doi.org/10.1016/j.bbrc.2012.12.043).

1.2 Running fastICA with icasso stabilisation

See the online tutorial *Running fastICA with icasso stabilisation* at: <https://urszulaczerwinska.github.io/DeconICA/icasso.html>

Running fastICA with icasso stabilisation

Setup your Matlab environment

Urszula Czerwinska

2018-04-28

- [Introduction fastICA with icasso stabilisation](#)
 - [What is fastICA?](#)
 - [What is icasso stabilisation](#)
 - [MSTD measure](#)
- [I have Matlab on my computer](#)
 - [Running fastICA with icasso stabilisation from deconICA](#)
 - [Running fastICA with icasso stabilisation directly in MATLAB](#)
 - [Running fastICA with icasso stabilisation in BIODICA](#)
- [I don't have Matlab on my computer](#)
 - [Running fastICA with icasso stabilisation in BIODICA docker image](#)
- [References](#)

Introduction fastICA with icasso stabilisation

What is fastICA?

Independent Components Analysis (ICA) is a Blind Source Separation (BSS) technique that aims to separate sources maximizing non-gaussianity (or minimising mutual information) of sources and therefore defining independent (or the most independent possible) components.

There exist many different implementations of ICA algorithm: Second Order Blind Identification (SOBI), Hyvarinen's fixed-point algorithm (FastICA), logistic Infomax (Infomax) and Joint Approximation Diagonalization of Eigenmatrices (JADE).

FastICA (Hyvärinen and Oja 2000) is a popular and fast implementation available in many programming languages.

1. Prewitthenning
 - a. Data centering

$$x_{ij} \leftarrow x_{ij} - \frac{1}{M} \sum_{j'} x_{ij'}$$

x_{ij} : data point

b. Whitenning

$$\mathbf{X} \leftarrow \mathbf{ED}^{-1/2} \mathbf{E}^T \mathbf{X}$$

Where - centered data, is the matrix of eigenvectors, is the diagonal matrix of eigenvalues

2. Single component extraction

a. Initialize w_i (in random)

b. $\mathbf{w}_i^+ \leftarrow E \{ \mathbf{X}g(\mathbf{w}_i^T \mathbf{X})^T \} \mathbf{w}_i - E \{ g'(\mathbf{w}_i^T \mathbf{X}) \}$

c. $\mathbf{w}_i \leftarrow \frac{\mathbf{w}_i^+}{\|\mathbf{w}_i^+\|}$

d. For $i = 1$, go to step g. Else, continue with step e.

e. $\mathbf{w}_i^+ \leftarrow \mathbf{w}_i - \sum_{j=1}^{j-1} w_i^T w_j w_j$

f. $\mathbf{w}_i \leftarrow \frac{\mathbf{w}_i^+}{\|\mathbf{w}_i^+\|}$

g. If not converged, go back to step b. Else go back to step a. with $i = i + 1$ until all components are extracted.

However, the results are not deterministic, as the w_i initial vector of weights is generated at random in the iterations of fastICA. If ICA is run multiple times, one can measure **stability** of a component. Stability of an independent component, in terms of varying the initial starts of the ICA algorithm, is a measure of internal compactness of a cluster of matched independent components produced in multiple ICA runs for the same dataset and with the same parameter set but with random initialization (Kairov et al. 2017).

What is icasso stabilisation

From (Himberg and Hyvärinen 2003):

We present an explorative visualization method for investigating the relations between estimates from FastICA. The algorithmic and statistical reliability is investigated by running the algorithm many times with different initial values or with differently bootstrapped data sets, respectively.

Resulting estimates are compared by visualizing their clustering according to a suitable similarity measure. Reliable estimates correspond to tight clusters, and unreliable ones to points which do not belong to any such cluster

Icasso procedure can be summarized in a few steps:

1. applying multiple runs of ICA with different initializations
2. clustering the resulting components
3. defining the final result as cluster centroids
4. estimating the compactness of the clusters

Icasso stabilisation is implemented in MATLAB. Despite our best effort we did not succeed to replicate this procedure in an open source language.

MSTD measure

In the version of fastica matlab package distributed with `deconICA` that we named `fastica++`, contains fasICA algorithm with default parameters, icasso stabilisation and MSTD index calculations and plots.

What is MSTD?

Most stable transcriptome dimension is a metric introduced in (Kairov et al. 2017).

Most Stable Transcriptome Dimension (MSTD) - ranking of independent components based on their stability in multiple ICA computation runs and define a distinguished number of components corresponding to the point of the qualitative change of the stability profile

This measure is not essential for `deconICA` as in the package we follow strategy of *overdecomposition*. However, it is useful to know the MSTD that as it is described in (Kairov et al. 2017) to characterize the data and estimate the numbers of components needed for *overdecomposition*.

Thus, we advise to use the matlab implementation of fastICA `fastica++` included in `deconICA` in order to enjoy the full functionalities of fastICA, icasso and MSTD.

I have Matlab on my computer

Running fastICA with icasso stabilisation from `deconICA`

Running matlab from `deconICA` is very easy. First, if you are not sure if you have matlab, you can run

```
matlabr::have_matlab()
```

```
## [1] TRUE
```

TRUE

If the answer is `TRUE` then you can run `run_fastICA()` function with `R=FALSE` and other parameters by default. Just as explained in the [tutorial](#).

FALSE

If the answer is `FALSE` but you are sure you have matlab installed, please find the path of your matlab executive. To find the path type `matlabroot` in your matlab session.

Then while running `run_fastica`, you simply provide the path in `matlbpth` parameter as in the example

My matlab defined in response to `matlabroot : /Applications/MATLAB_R2016a.app`.

```
#it is an example
```

```

library(deconica)
S <- matrix(runif(10000), 500, 5)
A <- matrix(runif(1000), 500, nrow = 5, byrow = FALSE)
X <- data.frame(S %*% A)
res <-
  run_fastICA(
    X = X,
    row.center = TRUE,
    n.comp = 5,
    overdecompose = FALSE,
    R = FALSE,
    matlabpth = "/Applications/MATLAB_R2016a.app/bin" #place your path + /bin here
  )

```

Running fastICA with icasso stabilisation directly in MATLAB

If you want to play with parameters of fastICA or you just prefer to use MATLAB directly, you can use a bunch of functions of `deconICA` to assure the smooth import of your results.

On an example of simulated matrix with 500 samples and 500 genes.

```

S <- matrix(runif(10000), 500, 5)
A <- matrix(runif(1000), 500, nrow = 5, byrow = FALSE)
X <- data.frame(S %*% A)
dim(X)

```

```
## [1] 500 500
```

```

colnames(X) <- paste0("S", 1:ncol(X))
row.names(X) <- paste0("gene_", 1:nrow(X))

```

Your data need to be centered and duplicated row names should be removed

```

X.pre <-
  prepare_data_for_ica(X, names = row.names(X), samples = colnames(X))

```

You can export your data into files saved on your disk.

You can add attribute name if you want to use a different name than the variable name (here 'X.pre\$df.scaled')

```

res.exp <- export_for_ICA(
  df.scaled.t = X.pre$df.scaled,
  names = X.pre$names,
  samples = X.pre$samples,
  n = 5
)

```

Then you can run in Matlab

```
cd 'path to fastica++'  
doICA(folder,fn,ncomp)
```

where

- folder is the path to the folder containing numerical **only** matrix
- fn is the file name containg the matrix
- n - number of components

here `cd './deconica/fastica++'`

```
doICA('/Users/xxxx/Documents/','X.pre$df.scaled_5_numerical.txt',5)
```

Then, if you followed the steps you can easily import the results.

```
res.imp <-  
  import_ICA_res(name = "X.pre$df.scaled",  
                  ncomp = 5,  
                  path_global_1 = "/Users/xxxx/Documents/")
```

`res.imp` object has then the fastICA results. You can add additional elements as the initial counts as to any R list object

```
res.imp$counts <- X
```

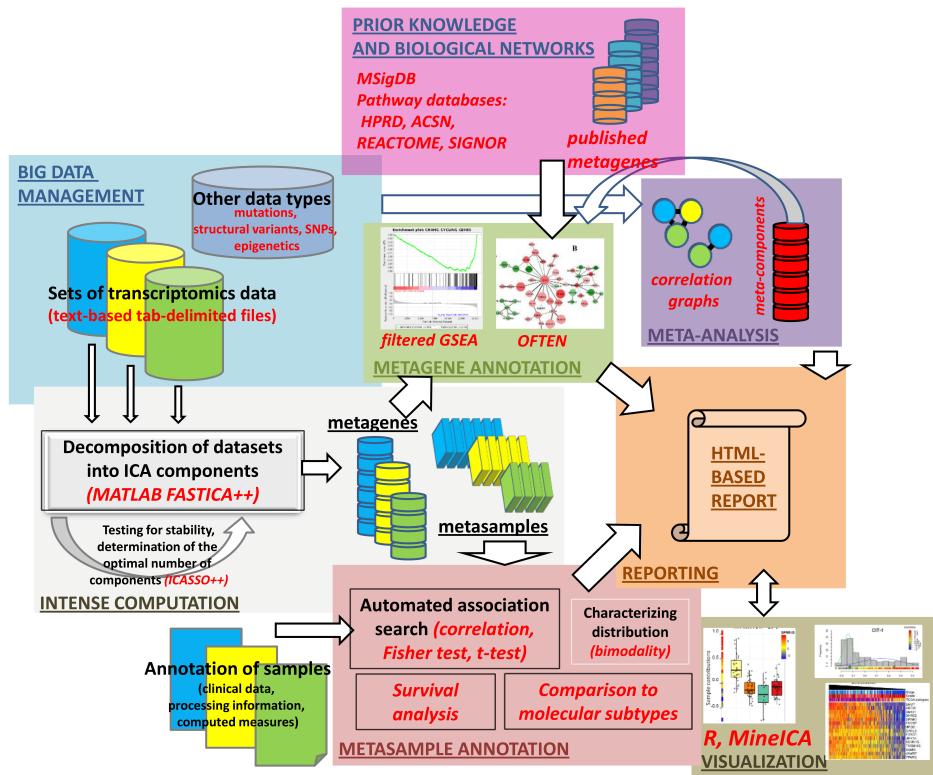
Running fastICA with icasso stabilisation in BIODICA

What is BIODICA

BIODICA is a computational pipeline implemented in Java language for

1. automating deconvolution of large omics datasets with optimization of deconvolution parameters,
2. helping in interpretation of the results of deconvolution application by automated annotation of the components using the best practices,
3. comparing the results of deconvolution of independent datasets for distinguishing reproducible signals, universal and specific for a particular disease/data type or subtype.

BIODICA framework focus is much larger than immune cells. It is quite complete user-friendly software whose applications and functions go beyond scope of this work (see following figure).



General architecture of the BIODICA data analysis pipeline. Boxes of different colors separates different functional modules of the system. Described functionality corresponds to the BIODICA version 1.0, source:

https://github.com/LabBandSB/BIODICA/blob/master/doc/ICA_pipeline_general_description_v0.9.pdf

The full BIODICA tutorial can be found [here](#).

BIODICA is essentially useful for general characterization of signal in transcriptomes.

Using BIODICA

Here we will focus on basic functions as running fastICA.

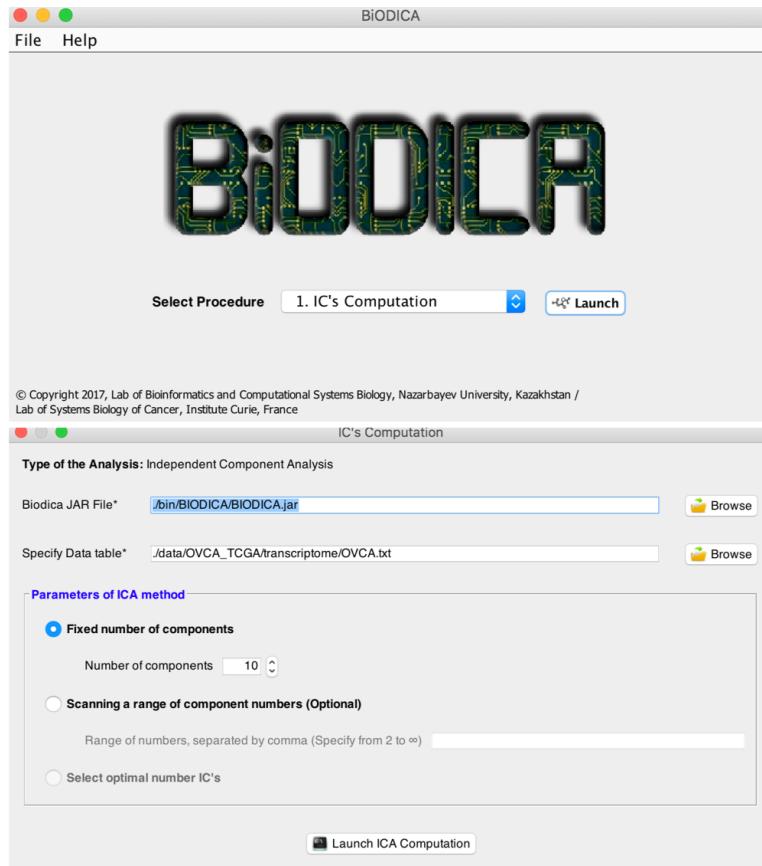
Requirements:

- Installed Java ver 1.6 or higher
- At least 8Gb of operating memory

In order to run an fastICA decomposition, one can use the GUI interface (see the figure below).

You can launch it by typing (or clicking on the file)

```
java -jar BIODICA_GUI.jar
```



Welcome screen of BIODICA with choice od funcitons and interface of fastICA data and parameters input

It is also possible to run it from command line.

This line will decompose `OVCA.txt` dataset into 20 components

```
java -jar BODICA.jar -config C:\Datas\BIODICA\config
-outputfolder C:\Datas\BIODICA\work\
-datatable C:\Datas\BIODICA\data\OVCA_TCGA\transcriptome\OVCA.txt -doicamatlab 20
```

The input to **BIODICA** is a dataset with genes in line and samples in columns with sample names and gene names, see [sample dataset](#). However if data is not in log, you need to put in log first. You should also eliminate the duplicated genes (not compulsory but advised for further interpretation). The data will be row centered by default by **BIODICA**.

I don't have Matlab on my computer

[Running fastICA with icasso stabilisation in BIODICA docker image](#)

1. Install [Docker](#) on your machine
2. Pull the biodica docker image

```
docker pull auranic/biodica
```

3. set `UseDocker = true` in the `config` file (in your cloned repo).

This procedure is also described on the [BIODICA wiki](#)

CONTACTS

All enquires about **BIODICA** state and development should be sent to

Andrei Zinovyev (<http://www.ihes.fr/~zinovyev>)

Ulykbek Kairov (https://www.researchgate.net/profile/Ulykbek_Kairov)

BIODICA software

BIODICA - computational pipeline for **I**ndependent **C**omponent **A**nalysis of **B**ig **O**mics **D**ata. It is a collaboration project between Lab of Bioinformatics and Systems Biology (Center for Life Sciences, Nazarbayev University, Kazakhstan) and Computational Systems Biology of Cancer Lab (Institute Curie, France). Principal Investigators and leading researchers of BIODICA Project: Andrei Zinovyev and Ulykbek Kairov.

References

- Himberg, Johan, and Aapo Hyvärinen. 2003. "ICASSO: Software for investigating the reliability of ICA estimates by clustering and visualization." In *Neural Networks for Signal Processing - Proceedings of the Ieee Workshop*, 2003-Janua:259–68. doi:[10.1109/NNSP.2003.1318025](https://doi.org/10.1109/NNSP.2003.1318025).
- Hyvärinen, Aapo, and Erkki Oja. 2000. "Independent Component Analysis: Algorithms and Applications." *Neural Networks* 13 (45): 411–30. doi:[10.1016/S0893-6080\(00\)00026-5](https://doi.org/10.1016/S0893-6080(00)00026-5).
- Kairov, Ulykbek, Laura Cantini, Alessandro Greco, Askhat Molkenov, Urszula Czerwinska, Emmanuel Barillot, and Andrei Zinovyev. 2017. "Determining the optimal number of independent components for reproducible transcriptomic data analysis." *BMC Genomics* 18 (1). doi:[10.1186/s12864-017-4112-9](https://doi.org/10.1186/s12864-017-4112-9).

1.3 Reference manual

The formal documentation of the R package automatically generated. Describes all functions and data with examples included in the package.

Package ‘deconica’

May 20, 2018

Type Package

Title Deconvolution of transcriptome through Immune Component Analysis

Version 0.1.0

Maintainer The package maintainer <urszula.czerwinska@cri-paris.org>

URL <https://github.com/UrszulaCzerwinska/DeconICA>

BugReports <https://github.com/UrszulaCzerwinska/DeconICA/issues>

Description Deconvolution of transcriptome through Immune Component Analysis aims to provide an analytical pipeline that can be applied to complex mixtures, i.e. transcriptomes in order to extract latent immune variables and provide a tool to study biological insights. It requires mixture data, additional data like .gmt for enrichment analysis and pure profiles of signals. It also allows simulation of gene expression data and comparison with other tools.

Depends R (>= 3.4)

License GPL

Encoding UTF-8

LazyData yes

Language en-US

RoxygenNote 6.0.1

Imports fastICA (>= 1.2-1),

stats (>= 3.4.1),

Hmisc (>= 4.0.3),

utils (>= 3.4.1),

gtools (>= 3.5.0)

Suggests edgeR (>= 3.18.1),

testthat,

pheatmap,

knitr,

rmarkdown,

CellMix,

prettydoc,

tableExtra,

analytics,

ACSNMiner (>= 0.16.8.25),

matlabr (>= 1.5.0),

ggplot2 (>= 2.2.1),

corrplot (>= 0.84),

reshape (>= 0.8.7),
 png (>= 0.1.7),
 grDevices (>= 3.4.1),
 NMF (>= 0.20.6),
 MCPcounter,
 Biobase,
 GEOquery,
 limma

biocViews**VignetteBuilder** knitr**R topics documented:**

add_path	3
assign_metagenes	3
BEK_ica_overdecompose	4
Biton.list	5
CAF.list	6
cell_voting_immgem	6
correlate_metagenes	7
deconica	8
dist_test_samples	9
doICA	10
doICABatch	11
Example_ds	12
export_for_correlation_java	13
export_for_ICA	14
generate_basis	15
generate_markers	16
gene_enrichment_test	17
get_matlab_2	18
get_max_correlations	19
get_scores	20
identify_immune_comp	21
ImmgenHUGO	21
import_ICA_res	22
is_logscale	23
LM22.list	23
lolypop_plot_corr	24
make_list	25
most_variant_IC	25
plot_dist_test	26
prepare_data_for_ica	27
radar_plot_corr	28
run_fastica	29
run_fastica_import	31
run_matlab_code_2	32
run_matlab_script_2	33
scores_corr_plot	33
simulate_gene_expressions	34
stacked_proportions_plot	35

<i>add_path</i>	3
TIMER_cellTypes	36
Index	37

<i>add_path</i>	<i>Create PATHs to add to MATLAB PATHs</i>
-----------------	--

Description

Create PATHs to add to MATLAB PATHs

Usage

```
add_path(path)
```

Arguments

path	path to add
------	-------------

Value

A character vector

Examples

```
add_path("~/")
```

<i>assign_metagenes</i>	<i>Assign components to a metagene through mutual reciprocity</i>
-------------------------	---

Description

Attributes labels to components under condition of mutual reciprocal correlation

Usage

```
assign_metagenes(corr, exclude_name = "M8_IMMUNE")
```

Arguments

corr	the correlation matrix, with at least r matrix and p matrix, can be generated from correlate_metagenes function
exclude_name	name of the components (present in r) to be excluded from this analysis (for example immune), by default "M8_IMMUNE" is excluded

Details

This function assign a component to a metagene/profile through verification if the component's the maximal correlation points to a given profile and if for this profile the maximal correlation points back the that component. In mathematical terms, given correlations between the set of profiles/metagenes $A = A_1, \dots, A_m$ and S components matrix $S = IC1, \dots, ICN$, if

$$S_i = argmax_i(corr(Aj, S))$$

and

$$A_j = argmax_j(corr(S_i, A))$$

Value

returns a data frame with component name in the first column and assigned profile/metagene name in second column

See Also

[get_max_correlations](#), [correlate_metagenes](#)

Examples

```
res_run_ica <- run_fastica (
  Example_ds,
  overdecompose = FALSE,
  n.comp = 5,
  with.names = TRUE
)
corr <- correlate_metagenes(
  S = res_run_ica$S,
  gene.names = res_run_ica$names)

assign_metagenes(corr)
```

Description

A dataset overdecomposed (into 100 components). Data were downloaded from GEO, then [run_fastica](#) using MATLAB algorithm with stabilization as applied.

Usage

`BEK_ica_overdecompose`

Format

a list containing

A A ICA matrix (sample scores)

S S ICA matrix (gene scores)

names gene names

samples sample names

counts raw counts (non centered)

log.counts log2 counts (non centered)

Details

Source: Bekhouche I, Finetti P, Adelaïde J, Ferrari A et al. High-resolution comparative genomic hybridization of inflammatory breast cancer and identification of candidate genes. PLoS One 2011 Feb 9;6(2):e16950. PMID: 21339811

Source

<https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE23720>

Biton.list

Array of all Metagenes

Description

list of metagenes

Usage

`Biton.list`

Format

list of 11 elements

Source

[http://www.cell.com/cell-reports/abstract/S2211-1247\(14\)00904-8](http://www.cell.com/cell-reports/abstract/S2211-1247(14)00904-8)

CAF.list	<i>CAF signatures from breast cancer</i>
----------	--

Description

Signatures of 4 subtypes of CAF

Usage

```
CAF.list
```

Format

list of 4 data.frames

Details

Tchou J, Kossenkov AV, Chang L, Satija C et al. Human breast cancer associated fibroblasts exhibit subtype specific gene expression profiles. BMC Med Genomics 2012 Sep 6;5:39. PMID: 22954256

Source

<https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE37614>

cell_voting_immgene	<i>Attribute cell type to a component</i>
---------------------	---

Description

From [gene_enrichment_test](#) result constructs a summary table counting percentage of a certain cell type attributed to a component. Works only with Immgen signatures

Usage

```
cell_voting_immgene(enrich, n = 10)
```

Arguments

enrich	enrichment results from gene_enrichment_test
n	n top results taken into account, 10 by default

Value

list of data.frame for each non NULL result of enrichment list from [gene_enrichment_test](#)

See Also

[gene_enrichment_test](#)

Examples

```

set.seed(123)
res_run_ica <- run_fastica (
  Example_ds,
  overdecompose = TRUE,
  with.names = TRUE
)
corr <- correlate_metagenes(
  S = res_run_ica$S,
  gene.names = res_run_ica$names)

assign <- assign_metagenes(corr)
immune_c<- identify_immune_comp(corr$r[, "M8_IMMUNE"], assign[, "component"], threshold = 0.1)

enrichment <- gene_enrichment_test(
  res_run_ica$S,
  res_run_ica$names,
  names(immune_c),
  alternative = "greater",
  p.adjust.method = "none",
  n = 50,
  n.consider = 100,
  p.value.threshold = 0.005
)

cell_voting_immgene(enrichment$enrichment)

```

correlate_metagenes *Correlate components with known ranked lists of genes*

Description

Components obtained, for example, with `run_fastica` can be characterized through correlation with known ranked list (metagenes or profiles), by default this function is using metagenes from Biton et al. (2015), Cell. It is using `rcorr` function for correlations

Usage

```
correlate_metagenes(S, gene.names, metagenes = Biton.list, threshold = -Inf,
  n.genes.intersect = 30, orient.long = TRUE, orient.max = FALSE, ...)
```

Arguments

S	S matrix of components
gene.names	list of gene names, needs to be of the same length as nrow of S, for ICA it is recommended to run <code>run_fastica</code> with <code>with.names = TRUE</code> to assure compatibility
metagenes	named list of datasets, each with two columns 1st - gene names, 2nd - ranks, by default 11 metagenes from Biton et al. (2015), Cell
threshold	threshold for components (columns of S) to be applied before correlation, default set to -Inf (all ranks are kept)

```

n.genes.intersect
    minimum of genes that should intersect between a component and a metagene
    to keep the component in correlation matrix

orient.long      orient by long tails, default TRUE
orient.max       orient by maximal correlation, default FALSE, can be used if there is no long
                  tails
...
    additional params you can pass to rcorr

```

Value

a correlation matrix with correlation coefficient r , p.values P and number of overlapping genes n , oriented S matrix

See Also

[rcorr](#) [run_fastica](#) [make_list](#)

Examples

```

res_run_ica <- run_fastica (
  Example_ds,
  overdecompose = FALSE,
  n.comp = 5,
  with.names = TRUE
)
correlate_metagenes(
  S = res_run_ica$S,
  gene.names = res_run_ica$names)

```

deconica

deconICA: Deconvolution of transcriptome through Immune Component Analysis

Description

deconICA is a package to perform unsupervised deconvolution of complex mixtures, it contains functions implementing the pipeline of data interpretation

Details

See the README on [CRAN](#) or [GitHub](#)

deconICA functions

NA

<code>dist_test_samples</code>	<i>Test impact of each Independent Component</i>
--------------------------------	--

Description

This function is applying distribution statistical test (i.e. `t.test`, `wilcox.test`) to evaluate which ICs have highest impact on differences between samples

Usage

```
dist_test_samples(A, sample.names, quant = c(0.1, 0.9), X.counts, test.type,
                 thr = 0.1, isLog = NULL, return = "p.value", wide = TRUE)
```

Arguments

A	result of <code>run_fastica</code> the A matrix
sample.names	names of samples, should correspond to number of columns of A
quant	quantiles to use, in form of <code>c(x, y)</code>
X.counts	expression data
test.type	test of distributions to perform
thr	threshold of maximal p.value considered 0.1 by default
isLog	by default NULL, if X is not counts but log, provide the base of log, for natural logarithm use <code>exp(1)</code>
return	if you want to return p.values select
wide	should the output matrix be in wide format (FALSE preferable for plotting)

Value

returns a matrix (in long or wide) format

Examples

```
# numerical matrix
set.seed(123)
S <- matrix(stats:::rnbinom(10000, mu = 6, size = 10), 500, 80)
dat <- matrix(runif(1600,min =1, max=10 ), 80, 80, byrow = TRUE)
A <- dat / rowSums(dat)
X <- data.frame(S %*% A)
res_run_ica <- run_fastica(X, row.center = TRUE, n.comp = 5, overdecompose = FALSE)

#stats:::t.test
dist_test_samples(A = res_run_ica$A,
                  sample.names = res_run_ica$samples,
                  X.counts = res_run_ica$log.counts,
                  test.type = "t.test",
                  isLog = 2,
                  return = "p.value",
                  thr= 0.5)

#edgeR::exactTest
```

```

dist_test_samples(A = res_run_ica$A,
sample.names = res_run_ica$samples,
X.counts = res_run_ica$log.counts,
test.type = "exactTest",
isLog = 2,
return = "p.value",
thr= 0.5)

#for plotting
res.ttest <- dist_test_samples(A = res_run_ica$A,
sample.names = res_run_ica$samples,
X.counts = res_run_ica$log.counts,
test.type = "t.test",
isLog = 2,
return = "p.value",
thr= 0.5,
wide = FALSE)

plot_dist_test(res.ttest, plot.type = "density")
plot_dist_test(res.ttest, plot.type = "line")

```

doICA

Call doICA matlab function

Description

function used inside [run_fastica](#) to run fastICA with icasso stabilization. Matlab engine is necessary

Usage

```
doICA(df.scaled.t, names, samples, path_global = getwd(), n, name = FALSE,
      export.corr = FALSE, corr_folder = "CORRELATION", matlbpth = NULL,
      fasticapth = paste0(path.package("deconica", quiet = TRUE), "/fastica++"))
```

Arguments

df.scaled.t	scaled numerical data matrix
names	gene names, no duplicates
samples	sample names
path_global	path where files will be saved
n	number of components
name	FALSE by default, name of dataset is used, you can put your name
export.corr	FALSE by default, if you want to use a java correlation function later or select TRUE
corr_folder	"CORRELATION" by default, only if you selected export.corr = TRUE
matlbpth	is found automatically with get_matlab_2 function, replace if not functional
fasticapth	path to fastica++ repository with MATLAB scripts

Value

it returns A, S matrices of ICA and names and samples for coherence

See Also

[get_matlab](#), [run_fastica](#), [export_for_ICA](#), [run_matlab_code](#), [import_ICA_res](#), [codeexport_for_correlation_java](#)

Examples

```
## Not run:
data(Example_ds)
res.pre <- 
  prepare_data_for_ica(Example_ds[, -1], names = Example_ds[, 1])
res.do <- doICA(
  df.scaled.t = res.pre$df.scaled,
  names = res.pre$names,
  samples = res.pre$samples,
  path_global = getwd(),
  n = 5,
  name = "test",
  export.corr = FALSE
)
## End(Not run)
```

doICABatch

doBatchICA

Description

prepares the data (scales and removes duplicates), runs `doBatchICA.m` MATLAB script

Usage

```
doICABatch(df, vec, path_global = getwd(), names, samples, name = FALSE,
  matlbpth = NULL, fasticapth = paste0(path.package("deconica", quiet =
  TRUE), "/fastica++"))
```

Arguments

df	numerical data matrix
vec	vector of values for which ICA should be computed
path_global	path were files will be saved, current directory by default
names	gene names
samples	sample names
name	name of the dataset, if not provided, name of R variable
matlbpth	path to matlab, found automatically with get_matlab_2
fasticapth	path to fastica++

Value

plots of stability and MSTD if possible

Examples

```
## Not run:
data(Example_ds)
doICABatch(
  Example_ds[, -1],
  seq(2, 4, 1),
  names = Example_ds[, 1],
  samples = colnames(Example_ds[, -1]),
  name = "test",
  fasticapth = paste0(path.package("deconica", quiet = FALSE), "/fastica++")
)
## End(Not run)
```

Example_ds

*Example of a cancer dataset***Description**

A a sample 60 randomly selected samples from transcriptome of inflammatory breast cancer (IBC). Data were centred and in transformed in log2 before sampling

Usage

Example_ds

Format

a dataframe with the

rows 21320

columns 61

first column is related to GENE names

Details

Bekhouche I, Finetti P, Adelaiide J, Ferrari A et al. High-resolution comparative genomic hybridization of inflammatory breast cancer and identification of candidate genes. PLoS One 2011 Feb 9;6(2):e16950. PMID: 21339811

Source

<https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE23720>

export_for_correlation_java
Exports S ICA matrix in a specific format

Description

needed for an external function in java

Usage

```
export_for_correlation_java(corr_folder = "CORRELATION", names, S, samples, A,
                            ncomp, name, path_global_1 = getwd())
```

Arguments

corr_folder	export folder name "CORRELATION" by default
names	gene names
S	S ICA matrix
samples	sample names
A	A ICA matrix
ncomp	number of computed components
name	name of the dataset
path_global_1	absolute path

Value

saves on the drive in corr_folder exported files

See Also

[run_fastica](#), [import_ICA_res](#), [doICA](#), [export_for_ICA](#)

Examples

```
## Not run:
data(Example_ds)
res.pre <-
  prepare_data_for_ica(Example_ds[, -1], names = Example_ds[, 1])
res.do <- doICA(
  df.scaled.t = res.pre$df.scaled,
  names = res.pre$names,
  samples = res.pre$samples,
  path_global = getwd(),
  n = 5,
  name = "test",
  export.corr = FALSE
)
export_for_correlation_java(
  S = res.do$S,
  A = t(res.do$A),
  names = res.do$names,
```

```

samples = res.do$samples,
name = "test",
ncomp = 5
)

## End(Not run)

```

export_for_ICA *Export files*

Description

export files in right format to run fastICA in MATLAB or BiodICA

Usage

```
export_for_ICA(df.scaled.t, names, samples, path_global = getwd(),
  name = FALSE, n = "")
```

Arguments

df.scaled.t	scaled numerical matrix
names	gene names, vector of character string
samples	sample names, vector of character string
path_global	path to export files, current directory by default
name	name of the dataset
n	number of components

Value

writes files on the drive in indicated location

See Also

[run_fastica](#), [import_ICA_res](#), [doICA](#), [export_for_correlation_java](#)

Examples

```

## Not run:
data(Example_ds)
res.pre <-
  prepare_data_for_ica(Example_ds[, -1], names = Example_ds[, 1])
export_for_ICA(res.pre$df.scaled,
  res.pre$names,
  res.pre$samples,
  path_global = getwd(),
  name ="test",
  n = 5)

## End(Not run)

```

<code>generate_basis</code>	<i>Generate basis matrix</i>
-----------------------------	------------------------------

Description

It generates a basis matrix that can be used for regression from list of weighted markers

Usage

```
generate_basis(df, sel.comp, markers, orient.long = TRUE)
```

Arguments

<code>df</code>	output of <code>run_fastica</code> containing at least <code>S</code> and <code>names</code> elements
<code>sel.comp</code>	components identified as specific sources (i.e. immune cells), by default it takes all components of <code>S</code> matrix, can be provided as valid column names or numeric index
<code>markers</code>	list of markers that should be used for basis matrix (i.e. "gene.list" from <code>generate_markers</code>), can be also simple vector or list of gene names
<code>orient.long</code>	TRUE by default, if you modified <code>S</code> matrix and you don't want it to be oriented select FALSE

Value

it returns a `data.frame` (basis matrix) that can be used for regression or visualization purposes

Examples

```
set.seed(123)
res_run_ica <- run_fastica (
  Example.ds,
  overdecompose = FALSE,
  n.comp = 20,
  with.names = TRUE
)
corr <- correlate_metagenes(
  S = res_run_ica$S,
  gene.names = res_run_ica$names)

assign <- assign_metagenes(corr)

immune <- identify_immune_comp(corr$r[,"M8_IMMUNE"], assign[, "component"], threshold = 0.1)

markers <- generate_markers(df = res_run_ica,n = 10,sel.comp= names(immune), return= "gene.list")
basis <- generate_basis(df = res_run_ica,sel.comp= names(immune),markers= markers )
pheatmap::pheatmap(basis )
```

generate_markers*Generate markers from components*

Description

It extracts from set of components (i.e. ICA S matrix) the n top genes (with weights if needed) to use as marker list or markers with weights for estimation of abundance through [get_scores](#)

Usage

```
generate_markers(df, n = 30, thr = Inf, sel.comp = paste("IC",
  1:ncol(df$S), sep = ""), return = "gene.list", orient.long = TRUE)
```

Arguments

df	list (usually output of run_fastica) containing at least S and names elements
n	number of top genes considered from each signature, n = 30 by default
thr	max gene expression, if removal of outliers is necessary, Inf (no threshold) by default.
sel.comp	components of interest (i.e. identified as specific to some profiles/metagenes (i.e. immune cells)), by default it takes all columns of S matrix, can be provided as valid column names or numeric index
return	return gene.list or gene.ranked
orient.long	TRUE by default, if S is oriented change to FALSE

Value

function returns either list of gene markers gene.list for each component or list of gene.ranked which are gene names with weights

See Also

[run_fastica](#), [get_scores](#)

Examples

```
set.seed(123)
res_run_ica <- run_fastica (
  Example.ds,
  overdecompose = FALSE,
  n.comp = 20,
  with.names = TRUE
)
corr <- correlate_metagenes(
  S = res_run_ica$S,
  gene.names = res_run_ica$names)

assign <- assign_metagenes(corr)

immune <- identify_immune_comp(corr$r[, "M8_IMMUNE"], assign[, "component"], threshold = 0.1)

generate_markers(df = res_run_ica, n = 10, sel.comp= names(immune))
generate_markers(df = res_run_ica, n = 10, sel.comp= names(immune), return= "gene.ranked")
```

gene_enrichment_test *Enrichment analysis*

Description

Computes an enrichment score (fisher exact test) in provided signatures for selected components

Usage

```
gene_enrichment_test(S, gene.names, immune.ics, gmt = ImmgenHUGO,
                     alternative = c("greater", "lower"), p.adjust.method = c("holm",
                     "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none"),
                     n = 100, n.consider = 500, min_module_size = 5, max_module_size = 500,
                     p.value.threshold = 0.05, orient.long = TRUE)
```

Arguments

S	matrix of components, dim n corresponding to genes, m corresponding to number of components, use oriented matrix
gene.names	character vector of gene names, length needs to be equal to n
immune.ics	vector of character names of components to use for enrichment test
gmt	data.frame obtained from gmt file with a function <code>format_from_gmt</code> , by default Immgen signatures http://Immgen.org
alternative	greater will check for enrichment, less will check for depletion
p.adjust.method	correction method
n	number of top genes that will be used to test signature
n.consider	number of genes from the positive end to be considered
min_module_size	minimal module size from gmt file to be considered in enrichment
max_module_size	maximum module size from gmt file to be considered in enrichment
p.value.threshold	maximal p-value (corrected if correction is enabled) that will be displayed
orient.long	TRUE by default, in case you applied transformation to your S components, select FALSE.

Details

`gene_enrichment_test` runs enrichment of a component (or any ranked list) in known (i.e. immune cell types) signatures. It was designed to use S matrix from `run_fastica_fisher.test` only on components identified as correlated with immune metagene through function `identify_immune_ic` and it searches in Immgen signatures <http://Immgen.org>.

Value

returns value if there is an enrichment in provided signatures:

metagenes interpreted metagene gene ranking

enrichment full results of the enrichment analysis sorted by corrected p.value

genes.list list of genes used for enrichment

See Also

[identify immune comp](#) identifying immune related components, [run_fastica](#) for running Independent Components Analysis, and [enrichment](#) for enrichment in gmt files

Examples

```
set.seed(123)
res_run_ica <- run_fastica (
  Example_ds,
  overdecompose = FALSE,
  n.comp = 41,
  with.names = TRUE
)
corr <- correlate_metagenes(
  S = res_run_ica$$S,
  gene.names = res_run_ica$names)

assign <- assign_metagenes(corr)

immune_c<- identify_immune_comp(corr$r[,"M8_IMMUNE"], assign[, "component"], threshold = 0.1)

gene_enrichment_test(
  res_run_ica$$S,
  res_run_ica$names,
  names(immune_c),
  alternative = "greater",
  p.adjust.method = "none",
  n = 50,
  n.consider = 100,
  p.value.threshold = 0.005
)
```

get_matlab_2

*Find matlab path***Description**

This tries to find matlab's path using a system which command, and then, if not found, looks at getOption("matlab.path"). If not path is found, it fails.

Usage

```
get_matlab_2(try_defaults = TRUE, desktop = FALSE, splash = FALSE,
  display = FALSE, wait = TRUE, mpath = NULL)
```

Arguments

try_defaults	(logical) If matlab is not found from Sys.which, and matlab.path not found, then try some default PATHs for Linux and OS X.
desktop	Should desktop be active for MATLAB?
splash	Should splash be active for MATLAB?

display	Should display be active for MATLAB?
wait	Should R wait for the command to finish. Both passed to system and adds the <code>-wait</code> flag.
mpath	path to matlab if known

Value

Character of command for matlab

Examples

```
if (matlabr::have_matlab()) {
  get_matlab_2()
}
```

`get_max_correlations` *Assign through maximal correlations*

Description

It assigns maximal correlations between set of correlated vectors

Usage

```
get_max_correlations(corr)
```

Arguments

corr	list of correlation matrices with correlation coefficients and p-values, can be obtained from correlate_metagenes or rcorr
------	--

Value

`data.frame` with matched column names, Pearson correlation coefficient, `p.value`

See Also

[rcorr](#), [correlate_metagenes](#), [assign_metagenes](#)

Examples

```
res_run_ica <- run_fastica (
  Example_ds,
  overdecompose = FALSE,
  n.comp = 5,
  with.names = TRUE
)
corr <- correlate_metagenes(
  S = res_run_ica$$,
  gene.names = res_run_ica$names)

get_max_correlations(corr)
```

<code>get_scores</code>	<i>Get abundance scores</i>
-------------------------	-----------------------------

Description

It calculates abundance scores through a mean of marker genes

Usage

```
get_scores(df, markers.list, summary = "mean", ...)
```

Arguments

<code>df</code>	gene matrix with samples in columns and genes in rows with named rows
<code>markers.list</code>	list of genes or list of genes with weights
<code>summary</code>	can be any type of mean i.e. <code>mean</code> , <code>gm_mean</code> (geometric mean), <code>harmonic_mean</code> , <code>weighted.mean</code> . For weighted mean weights are needed along with gene names
<code>...</code>	optional parameters for the mean function

Value

Function returns numerical value for each column (sample) of provided data frame

Examples

```
set.seed(123)
res_run_ica <- run_fastica (
  Example_ds,
  overdecompose = FALSE,
  n.comp = 20,
  with.names = TRUE
)
corr <- correlate_metagenes(
  S = res_run_ica$S,
  gene.names = res_run_ica$names)

assign <- assign_metagenes(corr)

immune <- identify_immune_comp(corr$r[, "M8_IMMUNE"], assign[, "component"], threshold = 0.1)
counts.abs <- (2^res_run_ica$log.counts)-1
row.names(counts.abs) <- res_run_ica$names

markers <- generate_markers(df = res_run_ica, n = 10,
                             sel.comp= names(immune),
                             return= "gene.list")
get_scores (counts.abs, markers, summary = "mean", na.rm = TRUE)

markers <- generate_markers(df = res_run_ica, n = 10,
                             sel.comp= names(immune),
                             return= "gene.ranked")
get_scores (counts.abs, markers, summary = "weighted.mean", na.rm = TRUE)
```

`identify_immune_comp` *Identify components related to immune signal*

Description

Identify components related to immune signal

Usage

```
identify_immune_comp(x, l, threshold = 0.1)
```

Arguments

x	the correlation with immune metagene can be retrieved from correlate_metagenes output
l	vector of names of assigned components
threshold	lower bound for filtering correlation [0,1]

Value

it returns data frame of component names and correlations passing the threshold

Examples

```
res_run_ica <- run_fastica (
  Example_ds,
  overdecompose = FALSE,
  n.comp = 20,
  with.names = TRUE
)
corr <- correlate_metagenes(
  S = res_run_ica$S,
  gene.names = res_run_ica$names)

assign <- assign_metagenes(corr)

identify_immune_comp(corr$r[, "M8_IMMUNE"], assign[, "component"], threshold = 0.1)
```

Description

Imported in correct format with [format_from_gmt](#) and parsed

Usage

ImmgenHUGO

Format

a dataframe with the
module first column
module length second column
gene names third column

Source

<http://www.immgen.org>

<i>import_ICA_res</i>	<i>Import results of ICA</i>
-----------------------	------------------------------

Description

imports files run in Matlab or precomputed

Usage

```
import_ICA_res(name, ncomp, path_global_1)
```

Arguments

name	name of the dataset
ncomp	number of components
path_global_1	absolute path of the files

Value

imports A and S ICA matrix

See Also

[run_fastica](#), [export_for_ICA](#), [doICA](#), [export_for_correlation_java](#)

Examples

```
## Not run:
data(Example_ds)
res.pre <-
  prepare_data_for_ica(Example_ds[, -1], names = Example_ds[, 1])
res.do <- doICA(
  df.scaled.t = res.pre$df.scaled,
  names = res.pre$names,
  samples = res.pre$samples,
  path_global = getwd(),
  n = 5,
  name = "test",
  export.corr = FALSE
)
import_ICA_res("test_5", 5, paste0(getwd(),"/test_5/"))

## End(Not run)
```

is_logscale	<i>Verify if data is in log scale</i>
-------------	---------------------------------------

Description

Verify if data is in log scale

Usage

```
is_logscale(x)
```

Arguments

x data.frame or matrix

Value

TRUE or FALSE

Examples

```
M <- matrix(sample(-1:14, 100, replace = TRUE), 10, 10, byrow = TRUE)
is_logscale(M)
M2 <- 2^M
is_logscale(M2)
```

LM22.list	<i>Array of all Metagenes</i>
-----------	-------------------------------

Description

list of 22 immune cell type profiles

Usage

```
LM22.list
```

Format

list of 22 data.frames

Source

[http://www.cell.com/cell-reports/abstract/S2211-1247\(14\)00904-8](http://www.cell.com/cell-reports/abstract/S2211-1247(14)00904-8)

<code>lolypop_plot_corr</code>	<i>Lolypop plot for correlations</i>
--------------------------------	--------------------------------------

Description

Plot correlations between one metagene or known profile and all components in a form of linear plot which is a variant of a signal plot. Wrapper using ggplot2.

Usage

```
lolypop_plot_corr(r, col, head.size = 10, head.color = "value",
  digits = 2, head.text.size = 3.5, head.text.color = "white",
  vertical = TRUE)
```

Arguments

<code>r</code>	correlation matrix <code>r</code> matrix of output correlate_metagenes
<code>col</code>	select column either index or column name
<code>head.size</code>	size of the point of correlation
<code>head.color</code>	by default colored by correlation values, if you want one color provide color name
<code>digits</code>	parameter of round for the correlation showed on the plot. integer indicating the number of decimal places (round) or significant digits (signif) to be used.
<code>head.text.size</code>	size of the correlation text font
<code>head.text.color</code>	color of the correlation text font
<code>vertical</code>	TRUE for vertical plot, FALSE for horizontal plot

Details

Values are order from highest correlation to lowest correlation. Colors and fonts can be overwritten.
To see all correlations simultaneously choose [radar_plot_corr](#)

Value

returns [ggplot](#)

See Also

[ggplot](#), [aes_string](#), [theme_bw](#), [geom_point](#), [labs](#), [scale_color_distiller](#), [coord_flip](#), [geom_segment](#)

Examples

```
res_run_ica <- run_fastica (
  Example_ds,
  overdecompose = FALSE,
  n.comp = 20,
  with.names = TRUE
)
corr <- correlate_metagenes(
  S = res_run_ica$S,
```

```

gene.names = res_run_ica$names)
#horizontal
lolypop_plot_corr(corr$r,2, vertical =FALSE)
# vertical
lolypop_plot_corr(corr$r,"M8_IMMUNE")
#change colors
lolypop_plot_corr(corr$r,"M8_IMMUNE",head.color = "black" , head.text.color = "green")
#remove title
lolypop_plot_corr(corr$r,"M8_IMMUNE")+ ggplot2::labs(title="",subtitle="")

```

make_list*Make list of weighted markers***Description**

Transforms a data frame with multiple columns into a named list of weighted markers with gene names in the first column and values in the second column.

Usage

```
make_list(df)
```

Arguments

df	data.frame to be transformed with gene names in the row.names
-----------	---

Value

named list of data.frames with gene names in the first column and values in the second column.

Examples

```

X <- as.data.frame(matrix(runif(10000), 50, 10))
row.names(X) <- paste("A",1:nrow(X), sep="")
make_list(X)

```

most_variant_IC*Compute variance explained by each Independent Component***Description**

Compute variance explained by each Independent Component

Usage

```
most_variant_IC(S, A, X, n = 5)
```

Arguments

S	result of run_fastica the S matrix
A	result of run_fastica the A matrix
X	data, either post-PCA data or run_fastica X matrix
n	number of top ICs if n = "all" then fraction of variance explained for all ICs is returned

Value

returns a data frame with n top ICs numbers ranked by their fraction of variance explained

Examples

```
set.seed(123)
res_fastica <- run_fastica (
  Example_ds,
  overdecompose = FALSE,
  n.comp = 20,
  with.names = TRUE
)
most_variant_IC(res_fastica$S, res_fastica$A, res_fastica$X, n =3)

res <- most_variant_IC(res_fastica$S, res_fastica$A, res_fastica$X, n =5)
barplot(as.matrix(t(res)))
```

plot_dist_test

Plot results of density test

Description

Wrapper over [ggplot](#) plotting either rank or density versus selected value in [dist_test_samples](#) (p.value or test statistics)

Usage

```
plot_dist_test(df, plot.type = c("line", "density"))
```

Arguments

df	data.frame in long format
plot.type	can be either "line" or "density"

Value

returns a line or density plot of p.value or test statistics versus rank or density

See Also

[ggplot](#), [stat_density](#), [theme_bw](#), [aes](#), [geom_line](#)

Examples

```
#numerical matrix
set.seed(134)
S <- matrix(stats::rnbino(m(10000, mu = 6, size = 10), 500, 80)
dat <- matrix(runif(1600,min =1, max=10 ), 80, 80, byrow = TRUE)
A <- dat / rowSums(dat)
X <- data.frame(S %*% A)
res_run_ica <- run_fastica(X, row.center = TRUE, n.comp = 5, overdecompose = FALSE)

#run the function selecting wide = FALSE
res.ttest <- dist_test_samples(A = res_run_ica$A,
sample.names = res_run_ica$samples,
X.counts = res_run_ica$log.counts,
test.type = "t.test",
thr=0.5,
isLog = 2,
return = "p.value",
wide = FALSE)

#plot results
plot_dist_test(res.ttest, plot.type = "density")
plot_dist_test(res.ttest, plot.type = "line")
```

`prepare_data_for_ica` *Formats data for ICA in MATLAB*

Description

Formats data for ICA in MATLAB

Usage

```
prepare_data_for_ica(df, names, samples = NULL, isLog = TRUE)
```

Arguments

<code>df</code>	numerical data matrix
<code>names</code>	gene names character vector
<code>samples</code>	if not provided column names will be used
<code>isLog</code>	are data in log? if FALSE data will be transformed to log2(x+1)

Value

<code>df.scaled</code>	scaled data without duplicates
<code>names</code>	gene names without duplicates
<code>non.scaled</code>	non scaled data without duplicates
<code>samples</code>	sample names

See Also

[run_fastica](#), [import_ICA_res](#), [doICA](#), [doICABatch](#)

Examples

```
data(Example_ds)
prepare_data_for_ica(Example_ds[, -1], names = Example_ds[, 1])
```

radar_plot_corr *Radar plot of correlations*

Description

Wrapper using ggplot2 to plot correlations between components and given metagenes or pure profiles

Usage

```
radar_plot_corr(df, ax.size = NULL, size.el.txt = 15, point.size = 5)
```

Arguments

<code>df</code>	output of function correlate_metagenes - correlation matrix with correlation and p-values
<code>ax.size</code>	define size of axis labels, adapts automatically by default
<code>size.el.txt</code>	define general size of letters, 15 by default
<code>point.size</code>	size parameter in geom_point

Value

Radar plots for correlations of each input component with matagene/profile, Returns a list containing the data.frame df used to generate the plot - long format - and the plot itself p.

See Also

[ggplot](#), [geom_point](#), [coord_polar](#), [theme_bw](#), [facet_wrap](#), [scale_color_distiller](#), [theme](#), [element_text](#)

Examples

```
res_run_ica <- run_fastica (
  Example_ds,
  overdecompose = FALSE,
  n.comp = 20,
  with.names = TRUE
)
corr <- correlate_metagenes(
  S = res_run_ica$S,
  gene.names = res_run_ica$names)

radar_plot_corr(corr)
data <- radar_plot_corr(corr)$df

#change plot
radar_plot_corr(corr)$p +
```

```
ggplot2::labs(title="11 Biton et al. metagenes vs my ICA components",
              subtitle="Pearson correlation coefficients")

radar_plot_corr(corr, point.size = 1)

radar_plot_corr(corr, point.size = 0)$p +
ggplot2::geom_point(size = 8, alpha = 0.4)
```

run_fastica

Decompose dataset with ICA.

Description

This is a wrapper of [fastICA](#). It allows compute number of ICs overdecompose for over decomposition for immune deconvolution.

Usage

```
run_fastica(X, overdecompose = TRUE, row.center = TRUE,
            with.names = FALSE, gene.names = NULL, samples = NULL,
            alg.typ = "parallel", method = "C", n.comp = 100, isLog = TRUE,
            R = TRUE, path_global = getwd(), matlbpth = NULL,
            fasticapth = paste0(path.package("deconica", quiet = TRUE), "/fastica++"),
            export.corr = FALSE, name = NULL, ...)
```

Arguments

X	a data matrix with n rows representing observations and p columns representing variables, place gene names in the first column and select with.names = TRUE
overdecompose	check TRUE to let select best number of components for deconvolution, for datasets >120 columns, n.comp will be set to 100, if <120 then number of components will be selected according to Kaiser Rule (90 percent of variance explained)
row.center	if TRUE subtract row mean from data
with.names	if first column of X is row.names please indicate TRUE, in case of duplicated names, the transcript with highest variance will be kept, names need to be HUGO names, if names are not provided at this step, you can provide them later
gene.names	character vector of row names - gene names
samples	if samples names different from column names
alg.typ	if alg.typ == "parallel" the components are extracted simultaneously (the default). if alg.typ == "deflation" the components are extracted one at a time.
method	if method == "R" then computations are done exclusively in R (default). The code allows the interested R user to see exactly what the algorithm does. if method == "C" then C code is used to perform most of the computations, which makes the algorithm run faster. During compilation the C code is linked to an optimized BLAS library if present, otherwise stand-alone BLAS routines are compiled.

<code>n.comp</code>	number of components to be extracted
<code>isLog</code>	if data is in log TRUE if data is in counts FALSE
<code>R</code>	if TRUE (default) the R version of fastICA is running, else the matlab version (you need to provide parameters of your matlab engine)
<code>path_global</code>	only if <code>R = FALSE</code> , the global path where files will be written, current directory by default
<code>matlbpth</code>	only if <code>R = FALSE</code> , the path to matlab engine, it uses <code>get_matlab</code> to find path to your matlab automatically
<code>fasticapth</code>	path to repository of source matlab code, it is set by default as coming with the package
<code>export.corr</code>	TRUE if you need to export S matrix in a specific format for correlation in external java app
<code>name</code>	important for Matlab version, defines the name of your files
<code>...</code>	other possible parameters for <code>fastICA</code>

Value

A list containing the following components as in `fastICA`

- `X` pre-processed data matrix (after PCA)
- `K` pre-whitening matrix that projects data onto the first `n.comp` principal components.
- `W` estimated un-mixing matrix (see definition in details)
- `A` estimated mixing matrix
- `S` estimated source matrix
- `names` if `with.names = TRUE` will contain row names list
- `counts` if `isLog = FALSE` will contain initial matrix without duplicated genes
- `log.counts` initial matrix without duplicated genes in $\log_2(x+1)$ before centering
- `samples` sample names as provided

See Also

`fastICA` <https://cran.r-project.org/web/packages/fastICA/index.html>

Examples

```
# numerical matrix
S <- matrix(runif(10000), 10, 2)
A <- matrix(sample(-3:3, 16, replace = TRUE), 2, 8, byrow = TRUE)
X <- data.frame(S %*% A)
run_fastica(X, row.center = TRUE, n.comp = 2, overdecompose = FALSE)
#matlab
## Not run:
run_fastica(X, row.center = TRUE, n.comp = 3, overdecompose = FALSE, R = FALSE)

## End(Not run)
# matrix with gene names
S <- matrix(runif(10000), 5000, 2)
A <- matrix(c(1, 1, -1, 3), 2, 2, byrow = TRUE)
X <- data.frame(S %*% A)
```

```

names <- paste("A", 1:nrow(X), sep="")
X <- cbind(names, X)
run_fastica(X, row.center = TRUE, n.comp = 2, overdecompose = FALSE, with.names = TRUE)

```

run_fastica_import *Reproduce process of run_fastica*

Description

Applies preprocessing of [run_fastica](#) but instead of running ICA it imports matlab output files. It is handy if you run the matlab idenpendly or if you lost R session data

Usage

```
run_fastica_import(X, overdecompose = TRUE, row.center = TRUE,
                   with.names = FALSE, gene.names = NULL, n.comp = 100, isLog = TRUE,
                   import = TRUE, path_global = getwd(), name = NULL, ...)
```

Arguments

X	a data matrix with n rows representing observations and p columns representing variables, place gene names in the first column and select with.names = TRUE
overdecompose	check TRUE to let select best number of components for deconvolution, for datasets >120 columns, n.comp will be set to 100, if <120 then number of components will be selected according to Kaiser Rule (90 percent of variance explained)
row.center	if TRUE subtract row mean from data
with.names	if first column of X is row.names please indicate TRUE, in case of duplicated names, the transcript with highest variance will be kept, names need to be HUGO names, if names are not provided at this step, you can provide them later
gene.names	character vector of row names - gene names
n.comp	number of components to be extracted
isLog	if data is in log TRUE if data is in counts FALSE
import	imports data only if TRUE
path_global	only if R = FALSE, the global path where files will be written, current directory by default
name	important for Matlab version, defines the name of your files
...	other possible parameters for fastICA

Value

an object as [run_fastica](#)

Examples

```
## Not run:

# numerical matrix
S <- matrix(runif(10000), 10, 2)
A <- matrix(sample(-3:3, 16, replace = TRUE), 2, 8, byrow = TRUE)
X <- data.frame(S %*% A)
#matlab
run_fastica(X, row.center = TRUE, n.comp = 2, overdecompose = FALSE, R = FALSE)
run_fastica_import(X, row.center = TRUE, n.comp = 2, overdecompose = FALSE, import=TRUE)

## End(Not run)
```

run_matlab_code_2 *Runs matlab code*

Description

This function takes in matlab code, where the last line must end with a ;, and returns the exit status, slightly modified version of [run_matlab_code](#)

Usage

```
run_matlab_code_2(code, matlbpth = NULL, endlines = TRUE, verbose = TRUE,
add_clear_all = FALSE, paths_to_add = NULL, ...)
```

Arguments

code	Character vector of code.
matlbpth	path to matlab engine
endlines	Logical of whether the semicolon (;) should be pasted to each element of the vector.
verbose	Print out filename to run
add_clear_all	Add clear all; to the beginning of code
paths_to_add	Character vector of PATHs to add to the script using add_path
...	Options passed to run_matlab_script

Value

Exit status of matlab code

Examples

```
if (matlabr:::have_matlab()){
  run_matlab_code_2("disp(version)", matlbpth = "matlbpth")
  run_matlab_code_2("disp(version)", paths_to_add = "~/")
  run_matlab_code_2(c("disp('The version of the matlab is:')", "disp(version)"))
  run_matlab_code_2(c("x = 5", "disp(['The value of x is ', num2str(x)]))")
}
```

<code>run_matlab_script_2</code>	<i>Run matlab script</i>
----------------------------------	--------------------------

Description

This function runs a matlab script, and returns exit statuses, slightly modified version of [run_matlab_script](#)

Usage

```
run_matlab_script_2(fname, matlbpth = NULL, verbose = TRUE,
                     desktop = FALSE, splash = FALSE, display = FALSE, wait = TRUE, ...)
```

Arguments

<code>fname</code>	Filename of matlab script (.m file)
<code>matlbpth</code>	path to matlab engine
<code>verbose</code>	print diagnostic messages
<code>desktop</code>	Should desktop be active for MATLAB?
<code>splash</code>	Should splash be active for MATLAB?
<code>display</code>	Should display be active for MATLAB?
<code>wait</code>	Should R wait for the command to finish. Both passed to system and adds the <code>-wait</code> flag.
<code>...</code>	Options passed to system

Value

Exit status of matlab code

<code>scores_corr_plot</code>	<i>Correlation plot of abundance scores</i>
-------------------------------	---

Description

Produces correlation plot of abundance scores estimated versus expected

Usage

```
scores_corr_plot(x, y, ...)
```

Arguments

<code>x</code>	matrix or data.frame of abundance scores, samples in rows and cell types in columns
<code>y</code>	matrix or data.frame of expected (or to compare) abundance scores, samples in rows and cell types in columns
<code>...</code>	additional parameters for method from corrplot

Details

correlation plot between different abundance scores of cell types in samples, correlates both matrices with each other merging two data.frames by row.names, on `corrplot` is.corr parameter is set to FALSE

Value

correlation plot based on `corrplot`
`corr.full` full correlation matrix
`corr.filtere` correlation without correlation with itself

See Also

`rcorr`, `corrplot`

Examples

```
x <- matrix(runif(1000), ncol = 10, nrow = 10)
y <- matrix(runif(1000), ncol = 10, nrow = 10)
row.names(x) <- row.names(y) <- paste0("S", 1:10)
colnames(x) <- paste0("CL_", 1:10, "_estimated")
colnames(y) <- paste0("CL_", 1:10, "_expected")
scores_corr_plot(x,y, method = "number", tl.col = "black")
scores_corr_plot(x,y, method = "square", tl.col = "black")
```

`simulate_gene_expresssion`
Simulate gene expression

Description

Function simulating gene expression of mixed cell types with a perturbator (i.e. proliferation, stress)

Usage

```
simulate_gene_expresssion(x, n, p, z = 0, dist.cells = list(dist =
  stats::rnbinom, size = 3, mu = 5), markers = NULL, mfold = 2,
  CLnames = NULL, genes = NULL, dist.noise.sources = list(dist =
  stats::rnorm, mean = 0, sd = 0.05), alpha = 1,
  dist.noise.global = list(dist = stats::rgamma, shape = 5, scale = 1),
  perturb = .pos.gaussian, pargs = list(p = p, mean = 0.5, sd = 0.2, lwr =
  0, upr = 1))
```

Arguments

<code>x</code>	number of cell types
<code>n</code>	number of genes
<code>p</code>	number of samples
<code>z</code>	number of perturbators
<code>dist.cells</code>	distribution and parameters from which cell profiles will be drawn

markers	number of markers that will distinguish cell types, can be a number (the same number of marker genes for cell types and perturbator), can be a vector of length x+z, it will be set to ceiling(n/20) if not provided
mfold	number of fold change between gene markers and other genes
CLnames	column names (cell and perturbator)
genes	gene names
dist.noise.sources	noise that will be added to each column of basis matrix (to each source)
alpha	parameter for the dirichlet distribution from which are drawn the cell proportions, using rdirichlet.
dist.noise.global	distribution and parameters of global noise (added to each sample one mixture is obtained)
perturb	function of distribution
pargs	arguments of perturbation function

Value

expression mixed expression matrix
marker.genes list of marker genes per cell type
basis_matrix pure cell type and perturbator profile
prop pure cell type and perturbator proportions (from 0 to 1)

Examples

```
res <- simulate_gene_expression (3, 30, 10, 2 , markers = c(4,5,5,3,4))
#visualise the basis matrix
pheatmap::pheatmap(res$basis_matrix)
#visualize expression
pheatmap::pheatmap(res$expression)
#observe distribution of signals
par(mfrow=c(2,2))
apply(res$basis_matrix, 2, hist)
```

stacked_proportions_plot
Plot cell proportions

Description

Plots scores for all samples as a fraction of one in each samples

Usage

```
stacked_proportions_plot(dat)
```

Arguments

dat	scores data.frame with cell types in lines and samples in columns
------------	---

Value

a stacked bar plot based on ggplot2

Examples

```
#random matrix y
y <- data.frame(matrix(runif(10000), ncol = 100, nrow = 10))
#plot
stacked_proportions_plot(y)
```

TIMER_cellTypes

TIMER signatures

Description

Signatures of 9 cell types published as part of TIMER tool

Usage

TIMER_cellTypes

Format

a dataframe with the

module first column

module length second column

gene names third column

Details

Li, Bo, et al. "Comprehensive analyses of tumor immunity: implications for cancer immunotherapy." *Genome biology* 17.1 (2016): 174.

Source

<http://cistrome.org/TIMER/>

Index

*Topic **datasets**
 BEK_ica_overdecompose, 4
 Biton.list, 5
 CAF.list, 6
 Example_ds, 12
 ImmgénHUGO, 21
 LM22.list, 23
 TIMER_cellTypes, 36

 add_path, 3, 32
 aes, 26
 aes_string, 24
 assign_metagenes, 3, 19

 BEK_ica_overdecompose, 4
 Biton.list, 5

 CAF.list, 6
 cell_voting_immgén, 6
 coord_flip, 24
 coord_polar, 28
 correlate_metagenes, 3, 4, 7, 19, 21, 24, 28
 corrplot, 33, 34

 deconica, 8
 deconica-package (deconica), 8
 dist_test_samples, 9, 26
 doICA, 10, 13, 14, 22, 27
 doICABatch, 11, 27

 element_text, 28
 enrichment, 18
 Example_ds, 12
 export_for_correlation_java, 11, 13, 14,
 22
 export_for_ICA, 11, 13, 14, 22

 facet_wrap, 28
 fastICA, 29–31
 fisher.test, 17
 format_from_gmt, 17, 21

 gene_enrichment_test, 6, 17
 generate_basis, 15
 generate_markers, 16

 geom_line, 26
 geom_point, 24, 28
 geom_segment, 24
 get_matlab, 11, 30
 get_matlab_2, 10, 11, 18
 get_max_correlations, 4, 19
 get_scores, 16, 20
 ggplot, 24, 26, 28

 identify_immune_comp, 18, 21
 ImmgénHUGO, 21
 import_ICA_res, 11, 13, 14, 22, 27
 is_logscale, 23

 labs, 24
 LM22.list, 23
 lolipop_plot_corr, 24

 make_list, 8, 25
 most_variant_IC, 25

 plot_dist_test, 26
 prepare_data_for_ica, 27

 radar_plot_corr, 24, 28
 rcorr, 7, 8, 19, 34
 round, 24
 run_fastica, 4, 7–11, 13, 14, 17, 18, 22, 26,
 27, 29, 31
 run_fastica_import, 31
 run_matlab_code, 11, 32
 run_matlab_code_2, 32
 run_matlab_script, 32, 33
 run_matlab_script_2, 33

 scale_color_distiller, 24, 28
 scores_corr_plot, 33
 simulate_gene_expressions, 34
 stacked_proportions_plot, 35
 stat_density, 26
 system, 19, 33

 theme, 28
 theme_bw, 24, 26, 28
 TIMER_cellTypes, 36

2 Publications and conferences

In this section I included publications with my minor contribution or not related to the thesis topic. At the end of the section, there is my CV with listed publications, conferences and selected graduated courses.

2.1 Adjustment of dendritic cells to the breast-cancer microenvironment is subset-specific

Paula Michea*, Floriane Noël*, Eve Zakine, **Urszula Czerwinska**, Philemon Sirven, Omar Abouzid, Christel Goudot, Alix Scholer-Dahirel, Anne Vincent-Salomon, Fabien Reyal, Sebastian Amigorena, Maude Guillot-Delost, Elodie Segura, and Vassili Soumelis

* contributed equally

Published in Nature Immunology on 16th July 2018

This project developed by Michea et al. originated in Vassili Soumelis group with interesting quality bulk RNA-seq data on pDC cells subsets in breast cancer.

On my side, I worked on an alternative to DGE (presented in this publication) approach aiming to verify if the pDC subsets can be discovered in an unsupervised manner from the data. As there were a number of samples for each subset available, I used ICA to decompose each subset. With the ICA components I created a correlation network of common and subset-specific signals.

On the other hand, I computed module activity scores with ROMA software [?] of each samples using a wide collection of pathways and then used hierarchical clustering to order the samples.

In my analysis, some subsets were clearly separated (MMAC, BDCA1pDC) and some not (BDCA1nDC and CD14pDC).

A strategical decision was taken to not to include my part of work in the main storyline. I actively participated in article writing and the review process.

Adjustment of dendritic cells to the breast-cancer microenvironment is subset specific

Paula Michea^{1,2,11}, Floriane Noël^{ID 1,2,3,11}, Eve Zakine^{1,2}, Urszula Czerwinska^{ID 1,2,4,5,6}, Philémon Sirven^{1,2}, Omar Abouzid^{1,2}, Christel Goudot^{1,2}, Alix Scholer-Dahirel^{1,2}, Anne Vincent-Salomon^{ID 7,8}, Fabien Reyal^{9,10}, Sebastian Amigorena^{1,2}, Maude Guillot-Delost^{ID 1,2}, Elodie Segura^{ID 1,2} and Vassili Soumelis^{ID 1,2*}

The functions and transcriptional profiles of dendritic cells (DCs) result from the interplay between ontogeny and tissue imprinting. How tumors shape human DCs is unknown. Here we used RNA-based next-generation sequencing to systematically analyze the transcriptomes of plasmacytoid pre-DCs (pDCs), cell populations enriched for type 1 conventional DCs (cDC1s), type 2 conventional DCs (cDC2s), CD14⁺ DCs and monocytes-macrophages from human primary luminal breast cancer (LBC) and triple-negative breast cancer (TNBC). By comparing tumor tissue with non-invaded tissue from the same patient, we found that 85% of the genes upregulated in DCs in LBC were specific to each DC subset. However, all DC subsets in TNBC commonly showed enrichment for the interferon pathway, but those in LBC did not. Finally, we defined transcriptional signatures specific for tumor DC subsets with a prognostic effect on their respective breast-cancer subtype. We conclude that the adjustment of DCs to the tumor microenvironment is subset specific and can be used to predict disease outcome. Our work also provides a resource for the identification of potential targets and biomarkers that might improve antitumor therapies.

Dendritic cells (DCs) are antigen-presenting cells (APCs) specialized in triggering adaptive immune responses through the activation of T cells¹. The various subsets of DCs have been defined on the basis of their ontogeny, phenotype and anatomical location^{2,3}. Advances in high-throughput technologies have improved the classification of DCs by identifying novel subset-specific markers and molecular signatures⁴. Studies of mice and human suggest that at steady state, ontogeny is a predominant factor in defining DC subset identity^{5–8}. For example, studies of plasmacytoid pre-DCs (pDCs)⁹ and the cDC1 and cDC2 subsets of conventional DCs (CD141⁺ DCs and CD11c⁺ DCs, respectively) from human blood and tonsils have revealed that pDCs cluster first by ontogeny independently of their tissue of origin¹⁰. In contrast, cDC1s and cDC2s are more sensitive to tissue localization, as tonsil cDC1s cluster closer to tonsil cDC2s than to blood cDC1s¹⁰. Tissue imprinting also influences DC function. Gut DCs induce the homing of T cells back to the gut through a mechanism dependent on retinoic acid, the chemokine receptor CCR9 and the integrin $\alpha_4\beta_7$, but spleen DCs do not¹¹. This suggests complex interplay between ontogeny and tissue imprinting, with the relative contribution of each remaining a matter of debate.

During inflammation, complex signals must be integrated by various DC subsets, which can change their function and molecular features^{12–17}. The diversity of DC subsets itself is also modified by inflammation through the appearance of monocyte-derived inflammatory DCs, which are absent in homeostatic conditions¹⁸. In humans, inflammatory DCs have been characterized in psoriatic skin^{19,20}, ascites fluid of ovarian cancer and synovial fluid of rheumatoid arthritis²¹. DCs infiltrate most cancer types. They serve

a protective role in anti-tumor immunity through the expression of co-stimulatory molecules and inflammatory cytokines and by inducing the activation of T cells^{22,23}. Conversely, DCs also promote immunosuppression by secreting anti-inflammatory cytokines^{24–27} or by expressing negative immunological checkpoint molecules, which inhibit T cell activation and are now being targeted by promising anti-tumor therapies^{28,29}. The plasticity of DCs in various tumor microenvironments (i.e., tissue imprinting), as well as specialized ontogeny-driven DC functions, might contribute to such molecular and functional heterogeneity.

In this study, we performed a systematic comparative transcriptomics study of DC subsets in human primary breast cancer and uninvolved tissue juxtaposed to the tumor, from the same patient. We found that the transcriptional reprogramming of tumor-infiltrating DCs was DC subset specific, suggestive of complex interplay between ontogeny and tissue imprinting in conditioning DC diversity in the tumor microenvironment. Our results also provide high-quality large-scale datasets of primary tumor-infiltrating DCs that constitute a valuable resource for the biomedical community.

Results

Phenotypically distinct APCs infiltrate human breast cancer. DCs that had infiltrated breast cancer tissues were identified by multicolor flow cytometry on the basis of published studies of human DC subsets²⁰. Because this was the first in-depth characterization of DC subsets in human breast cancer, to our knowledge, we performed preliminary analyses to validate our strategy. After standard gating to eliminate debris, doublets and dead cells, we selected CD45⁺ cells to efficiently exclude CD45⁻ cells, which were mainly

¹Institut Curie, PSL Research University, Paris, France. ²INSERM, UMR 932, F-75005 Paris, France. ³Université Paris Sud, Université Paris-Saclay, Orsay, France. ⁴INSERM, U900, F-75005 Paris, France. ⁵Mines ParisTech, Paris, France. ⁶Centre de Recherches Interdisciplinaires, Paris Descartes, Paris, France. ⁷Institut Curie, Service de Pathologie, Paris, France. ⁸INSERM, U934, F-75005 Paris, France. ⁹INSERM, UMR 932, Département de Recherche Translationnelle, Residual Tumor & Response to Treatment Laboratory (RT2Lab), Paris, France. ¹⁰Institut Curie, Département de Chirurgie, Paris, France.
¹¹These authors contributed equally: Paula Michea, Floriane Noël. *e-mail: vassili.soumelis@curie.fr

Table 1 | Variables that influence the disease-free survival of patients with breast cancer

	LBC		TNBC	
	HR	P value	HR	P value
pDC				
NPI > 5.4	1	-	-	-
NPI ≤ 5.4	0.31	$7.5 \times 10^{-13}*$	-	-
High signature ratio	1	-	-	-
Low signature ratio	1.37	0.0072*	-	-
cDC2				
NPI > 5.4	1	-	-	-
NPI ≤ 5.4	0.3	$1.8 \times 10^{-13}*$	-	-
High signature ratio	1	-	-	-
Low signature ratio	1.27	0.041*	-	-
cDC1e				
NPI > 5.4	1	-	1	-
NPI ≤ 5.4	0.29	$7.6 \times 10^{-14}*$	0.27	$1.1 \times 10^{-9}*$
High signature ratio	1	-	1	-
Low signature ratio	1.39	0.0041*	1.76	0.0058*
MonoMac				
NPI > 5.4	1	-	1	-
NPI ≤ 5.4	0.31	$5.9 \times 10^{-13}*$	0.28	$3.9 \times 10^{-9}*$
High signature ratio	1	-	1	-
Low signature ratio	0.77	0.025*	0.67	0.049*

Multivariate Cox regression analysis of predictors of disease-free survival that influence the disease-free survival of patients with LBC or TNBC (top), showing the Nottingham prognostic index (NPI) and 'subset-specific signature z-score' for each cell type (left column), with the hazard ratio (HR) and P value for each (*P < 0.05, Cox model likelihood test).

tumor cells and fibroblasts (Supplementary Fig. 1a). We used a panel of lineage markers (Lin) to exclude CD3+ T cells, CD19+ B cells and CD56+ cells (Supplementary Fig. 1a). We analyzed expression of the co-receptor CD14 independently of the lineage channel to efficiently identify CD14+ DCs, which have been reported in patients with cancer^{20,21,30–32}. Among Lin- cells, we next gated on CD11c+HLA-DR^{hi} cells to exclude CD11c+HLA-DR^{neg-lo} myeloid-derived suppressor cells³³. HLA-DR+CD123+ pDCs were identified in the CD11c- gate (Supplementary Fig. 1a).

In the Lin-CD45+ gate, we identified four distinct CD11c+ cell populations defined by their expression of the antigen-presenting molecule CD1c and CD14 (Fig. 1a). On the basis of published standardized nomenclature for blood DC subsets³⁴, CD1c+CD14- cells matched the definition of cDC2s, the CD1c-CD14- cell population included cDC1s, and CD1c-CD14+ cells were monocytes-macrophages (called 'MonoMacs' here) (Fig. 1a). We also identified a CD1c+CD14+ cell population that co-expressed markers of monocytes and macrophages, such as CD14, CD64 and CD163, and cDC2 markers, such as CD1c, CD206 and FcεRI (Fig. 1b and Supplementary Fig. 1b). Because these CD1c+CD14+ cells were phenotypically distinct from MonoMacs, and because they had not been systematically distinguished in published studies³⁴, we call them 'CD14+ DCs' here. CD56+CD14+ cells were reported to be interferon-producing killer DCs in the context of cancer³⁵.

and were subsequently shown to correspond to activated natural killer cells³⁶. A similar CD56+CD14+ phenotype has been described for fraction of blood monocytes from healthy donors³⁷. We detected CD56+CD14+ cells in breast-cancer samples (18% of live CD45+CD3-CD19- cells) (Supplementary Fig. 1c). Because of their controversial nature, we excluded them through the use of antibody to the adhesion molecule CD56 in our 'lineage cocktail' (Supplementary Fig. 1c).

The C-type lectin-like receptor Clec9A could not be used to identify cDC1s, as it was degraded during enzymatic digestion of the tissue (Supplementary Fig. 1d). Thrombomodulin (CD141 or BDCA3) was expressed promiscuously by all DCs, including pDCs and MonoMacs (Fig. 1b). However, CD141^{hi} cells were found only in the CD1c-CD14- population (Fig. 1b); hence, the CD1c-CD14- population showed considerable enrichment for cDC1s. Because CD141^{hi} cells were too few in number (<100 cells per sample) and rare (5–50% of CD141^{hi} cells among CD1c-CD14- cells in only half of the patients) to allow further separation into subsets, we designated the CD1c-CD14- cell subset 'cDC1-enriched' (cDC1e) cells and used it for further molecular characterization. MonoMacs, CD11c+HLA-DR^{neg-lo} cells, CD14+ DCs, cDC2s and cDC1e cells did not express the natural killer cell receptor CD16 (FcγRIII) (Fig. 1b and data not shown). MonoMacs, CD14+ DCs and cDC2s had high expression of CD32B, which has been reported on a non-inflammatory subset of cDC2s in the blood³⁸, but cDC1e cells did not. The receptor tyrosine kinase AXL, which is expressed by precursors of blood DCs and by cDC2s, was expressed mainly by cDC2s, CD14+ DCs and MonoMacs in breast tumors (Fig. 1b). This indicated a clear discrepancy between blood and breast tissue in terms of DC markers.

To investigate the morphology of tumor APCs, we sorted them and analyzed their cytological features. pDCs had a typical plasmacytoid morphology⁹, while cDC2s, cDC1e cells and CD14+ DCs had a dendritic morphology with high ratio of nucleus to cytoplasm and a less-basophilic cytoplasm than that of pDCs (Fig. 1c). MonoMacs had an acidophilic cytoplasm with abundant vacuoles (Fig. 1c), as is commonly observed in this population.

We quantified the distinct APC subsets across 22 luminal breast cancer (LBC) samples. MonoMacs were the most abundant cells (6.1% (median value) of CD45+ cells), followed by CD14+ DCs, and pDCs (0.5% and 0.3%, respectively, of CD45+ cells). cDC1e cells and cDC2s were the least abundant APCs (0.2% of CD45+ cells) (Fig. 1d). This phenotypic analysis identified and quantified five APC populations that infiltrated human breast cancer: MonoMacs, cDC2s, CD14+ DCs, pDCs and cDC1e cells.

Tumor-infiltrating DCs show enrichment for human blood DC signatures. Because the number of APCs obtained from primary breast cancer samples after sorting was very low (range, 2–12,000 cells), we adapted a protocol aimed at obtaining robust transcriptomes from rare cell populations by RNA-based next-generation sequencing (RNA-seq) (Supplementary Fig. 1e). We analyzed only those cell populations with more than 100 events. We generated RNA-seq profiles for pDCs, cDC2s, cDC1e cells, CD14+ DCs and MonoMacs from 13 patients with LBC (Supplementary Table 1), with 44 transcriptomes passing all quality-control criteria (Supplementary Table 2). On average, 60.5% of reads were mapped to the reference transcriptome across all samples. After filtering and normalizing the raw RNA-seq data, we obtained an average of 14,417 expressed genes.

To verify the identity of each of the subsets at the RNA level relative to the flow-cytometry analysis, we checked the expression of genes encoding various subset-specific and shared DC markers (Fig. 1e). As expected, pDCs had high expression of IL3RA, CLEC4C and TLR9; cDC2s had high expression of CD1A, CD1B and FCER1A; CLEC9A, XCR1 and BATF3 (all markers of cDC1s)

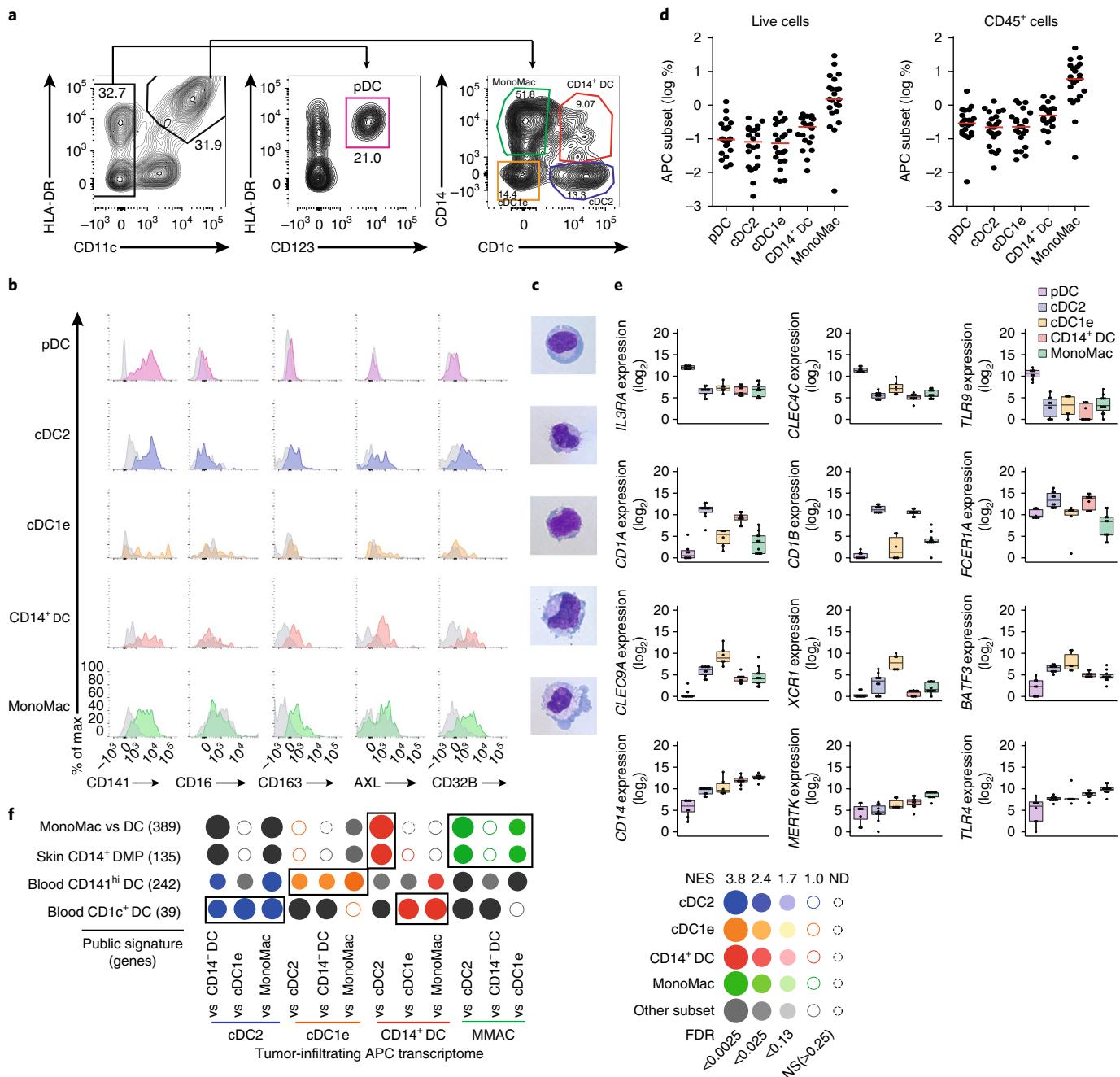


Fig. 1 | Phenotypic and molecular characterization of innate APCs that infiltrate breast cancer tissue. **a**, Flow cytometry showing the gating strategy used to distinguish DC subsets from MonoMacs in breast-cancer samples. Numbers in or adjacent to outlined areas indicate percent cells in each (subset designations included in plots here). **b**, Expression of CD141, CD16, CD163, AXL and CD32B by various APC subsets (left margin) in breast-cancer samples ($n=3$ donors, with similar results), presented as mean fluorescent intensity. **c**, Microscopy of Giemsa-stained cytopsin preparations, showing the morphology of APCs sorted by flow cytometry from tumors ($n=3$ donors, with similar results). Original magnification, $\times 100$. **d**, Frequency of APC subsets (horizontal axis) among total live cells (left) or $CD45^+$ cells (right), assessed by flow cytometry. Each symbol represents an individual donor ($n=22$); small red horizontal lines indicate the median. **e**, Expression of genes encoding DC-selective markers, by APCs (key) isolated from tumors, presented as read counts + 1 (\log_2 values). Each symbol represents an individual sample ($n=8$ (pDC), $n=10$ (cDC2), $n=6$ (cDC1e), $n=9$ (CD14⁺ DC) and $n=11$ (MonoMac)); middle line indicates the median, box limits indicate the first and third quartiles, and 'whiskers' indicate 'extreme' data points no more than 1.5x the length of the box beyond the box limit. **f**, Enrichment for various APC public signatures⁵⁹ (left margin) in pairwise comparisons (below plot) of the transcriptomes of APCs (key color) isolated from tumors (n values as in **e**), plotted with the BubbleMap module of BubbleGUM software; color intensity and symbol size indicate the normalized enrichment score (NES) and FDR, respectively (key at right); outlined areas indicate the expected signature-enrichment analysis. DMP, dermal mononuclear phagocyte.

were 'preferentially' expressed by cDC1e cells; MonoMacs had high expression of *CD14*, *MERTK* and *TLR4*; and CD14⁺ DCs shared the expression of *FCER1A* and *CD14* with cDC2s and MonoMacs,

respectively (Fig. 1e). Gene set-enrichment analyses using public datasets indicated that breast-cancer cDC2s had the highest normalized enrichment score (NES) with the blood cDC2 signature;

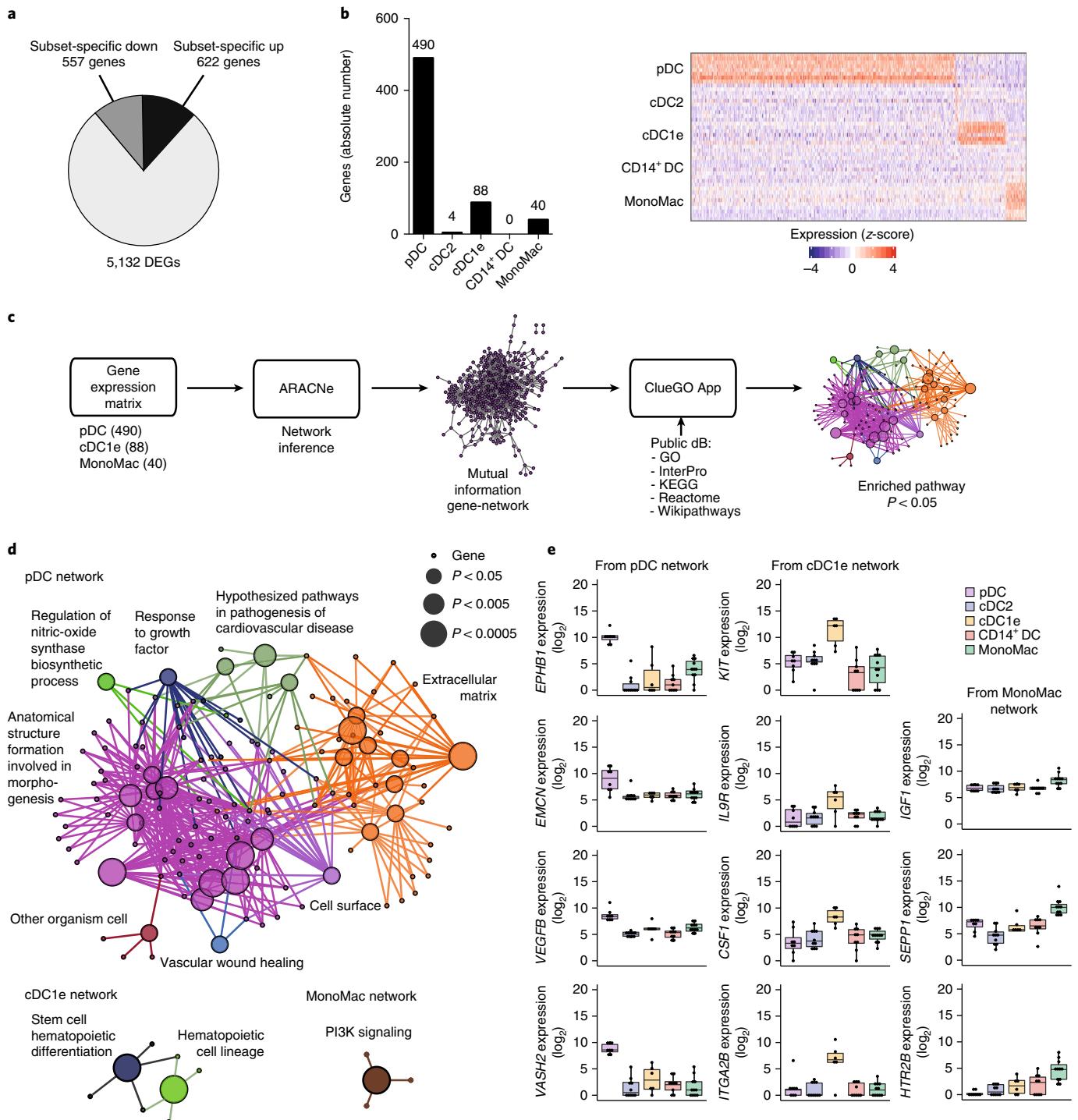


Fig. 2 | Subset-specific signatures that define tumor APCs. **a**, Proportion of genes upregulated (up) or downregulated (down) among DEGs in APC subsets isolated from tumors. $P < 0.05$ (one-way analysis of variance (ANOVA) and Tukey's post-hoc correction). **b**, Quantification of genes upregulated in one subset (horizontal axis) relative to their expression in all other subsets (left), and expression (z-score; key) of each gene (one per column) in the APC subsets (right margin; one sample per row) (right). Numbers above bars (left) indicate specific values. **c**, Bioinformatics pipeline used for functional inference (in **d**) from subset-specific gene signatures; numbers below plot at far left indicate the number of genes in each group. ARACNe, algorithm for the reconstruction of accurate cellular networks; ClueGO App, plug-in for the Cytoscape software platform for visualizing complex networks; dB, database (GO, gene ontology knowledgebase; InterPro, database of protein families, domains and functional site; KEGG, Kyoto encyclopedia of genes and genomes; Reactome, pathway database; Wikipathways, database of biological pathways). **d**, Functional network inference showing the biological pathways (along perimeter) most significantly overrepresented (FDR < 0.05) in the gene signatures of pDCs, cDC1e cells and MonoMacs (n values as in **e**) from tumors: color indicates pathway; node size indicates P value; smallest symbols indicate the pathway-associated genes (key). PI3K, phosphatidylinositol-3-OH kinase. **e**, Expression of genes in the pathways most significantly enriched in **d**, for pDCs ($n=8$ samples), cDC1e cells ($n=6$ samples) and MonoMacs ($n=11$ samples) (above plots), presented as in Fig. 1e.

breast-cancer cDC1e cells had the highest NES with the blood cDC1 (CD141^{hi}) signature; and breast-cancer MonoMacs had the highest NES with the CD14⁺ dermal mononuclear phagocyte and MonoMac signatures, compared with all the other gene signatures. Finally, breast-cancer CD14⁺ DCs shared the highest NES with the blood cDC and MonoMac skin CD14⁺ dermal mononuclear phagocyte signatures (Fig. 1f). Hence, robust transcriptional profiles confirmed the identity of the main DC subsets and MonoMacs that infiltrated breast cancer.

Tumor-infiltrating DC harbor subset-specific signatures. We performed analysis comparing gene expression in pDCs, cDC2s, cDC1e cells, CD14⁺ DCs and MonoMacs and identified 5,132 genes that were expressed differentially in at least one subset relative to their expression in all other APCs ('differentially expressed genes' (DEGs)) (Fig. 2a). We then applied a post-hoc test to extract the genes upregulated in each type of APC relative to their expression in all other subsets, which we defined as its subset-specific signature. From a total of 662 subset-specific genes, 490 corresponded to pDCs, 88 corresponded to cDC1e cells, 40 corresponded to MonoMacs and 4 corresponded to cDC2s (Fig. 2b). We found no genes specifically upregulated in CD14⁺ DCs (Fig. 2b).

Among the ten DEGs with the highest significance, genes encoding the oncoprotein TCL1A and the anti-apoptotic molecule ZFAT were found in the pDC signature; genes encoding the glutamate receptor GRIP and the cytokines CCL22 and IL-29 (*IFNL1*) were found in the cDC2 signature; genes encoding the plasma-membrane receptors IL-33R (ST2; *IL1RL1*) and XCR1 were found in the cDC1e cell signature; and genes encoding the fatty acid-biosynthesis enzymes ASAHI and ME1 were found in the MonoMac signature (Supplementary Table 3).

We then identified functions linked to each subset-specific signature (Fig. 2c,d). From a total of 29 pathways (false-discovery rate (FDR), <0.05) found in the pDC gene network, the most significantly enriched pathway was 'anatomical structure involved in morphogenesis' (FDR = 2.7×10^{-7}), and this included *EPHB1*, *VEGFB* and *VASH2* (Fig. 2e,f). The cDC1e cell gene network showed enrichment for two pathways, both linked to hematopoiesis, and this included *KIT*, *IL9R*, *CSF1* (which encodes the cytokine M-CSF) and *ITGA2B* (Fig. 2e,f). The MonoMac gene signature showed enrichment for only the pathway 'PI3K signaling', which included *IGF1*, *SEPP1* and *HT2RB* (Fig. 2e,f). Thus, subset-specific genes were identified for LBC-infiltrating pDCs, cDC2s, cDC1e cells and MonoMac. Notably, none of those subsets showed differential enrichment for any pathway directly linked to immunological function.

DC plasticity in the tumor microenvironment is subset specific. To determine how tumor-infiltrating APCs adapt to their microenvironment, we analyzed non-malignant tissue juxtaposed to tumor tissue ('juxta-tumoral' tissue) from eight donors (pairing tumor tissue with juxta-tumoral tissue from the same patient). The pDC, cDC2, cDC1e, CD14⁺ DC and MonoMac populations that we described in the tumors were also identified in the juxta-tumoral tissue, but with a lower frequency among the CD45⁺ cells than in the tumor, a result that was significant for pDCs ($P=0.078$) and cDC1e cells ($P=0.039$ (likelihood ratio test); Fig. 3a and Supplementary Fig. 2a). We generated transcriptional profiles for each APC subset in the juxta-tumoral tissue using the RNA-seq workflow used for the tumor DC subsets; the transcriptomes were generated in parallel, were run in the same batch as their tumoral counterpart and were matched for each donor (Supplementary Fig. 2b). We compared the transcriptome of each APC subset in the tumor with that of the juxta-tumoral sample (Supplementary Fig. 2b). We identified 607 DEGs for pDCs, 348 DEGs for CD14⁺ DCs, 236 DEGs for MonoMacs, 45 DEGs for cDC1e cells and 22 DEGs for cDC2s, which resulted in a total of 1,258 DEGs (FDR < 0.05, and

a change in expression of over onefold (\log_2 values)) that were used for further analysis (Fig. 3b). DEGs from all DC subsets had higher expression in the tumor than in the juxta-tumoral tissue (Fig. 3b). We identified seven genes with the highest significance (FDR = 1.72×10^{-17} to 4.1×10^{-10}) among DEGs of tumor CD14⁺ DCs and juxta-tumoral CD14⁺ DCs compared with DEGs from other APC subsets; these included genes encoding the secretoglobulins TFF1 and TFF3, which have a function in mucosal healing. Conversely to DCs, DEGs from MonoMacs were upregulated mostly in juxta-tumoral samples (195 DEGs) rather than tumor samples (41 DEG). Among the genes most significantly upregulated in juxta-tumoral MonoMacs was the gene encoding the scavenger receptor ligand CD163L, which is associated with M2 polarization (Fig. 3b).

Among the five transcripts whose expression was the most increased in tumor APCs relative to their expression in juxta-tumoral APCs ('top five'), we detected transcript encoding the negative regulator CD5 in pDCs (Fig. 3c) and transcripts encoding the secretoglobulins SCGB2A2 and SCGB1D2 in cDC2s. SCGB2A2 was also among the top five DEGs of CD14⁺ DCs and pDCs in the tumor-versus-juxta-tumoral comparison (Fig. 3c and Supplementary Fig. 2b). The gene encoding TACI (*TNFRSF13B*), a member of the cytokine TNF receptor superfamily, was among the top five DEGs upregulated in tumor cDC1e cells relative to their expression in juxta-tumoral cDC1e cells, whereas the gene encoding the chemokine CCL7 was substantially upregulated in tumor MonoMacs relative to its expression in juxta-tumoral MonoMacs (Fig. 3b). The gene encoding AGR2 (a protein disulfide isomerase needed for mucin folding) was among the genes with the most significant upregulation in tumor cDC2s, CD14⁺ DCs and MonoMacs relative to their expression in the juxta-tumoral counterparts of those cells (Fig. 3b).

We next analyzed whether the genes expressed differentially by tumor APCs relative to their expression in juxta-tumoral APCs were shared across subsets. Strikingly, most of the genes were expressed differentially exclusively in one subset (1,074 genes) or two subsets (184 genes) (Fig. 3d). Only 21 DEGs were shared with two other subsets and none were shared with three or four other subsets (Fig. 3d,e). This indicated that the tumor-induced transcriptional reprogramming of APCs was subset specific.

The differential expression of SCGB2A2, a gene previously associated with mammary epithelial tumor cells^{39,40}, raised questions about its tumor specificity versus its immune-cell specificity⁴¹. We excluded the possibility of contamination by tumor-cell mRNA by our stringent gating strategy (Fig. 1a and Supplementary Fig. 1) and by the observation that epithelium-specific mRNA, such as *EPCAM*, was not detected among the DEGs in tumor pDCs (Supplementary Fig. 2b). Given that SCGB2A2 was detected in a transcriptome analysis of blood pDCs from healthy donors⁴², these observations suggested that pDCs might express SCGB2A2 mRNA endogenously at steady state and in inflammatory conditions. These observations indicated that DCs adapted to the tumor microenvironment in a subset-specific manner.

Immunological pathways are absent from APC 'tumor-emerging genes'. For each APC separately, we analyzed the functions linked to molecules encoded by 'tumor-emerging genes' (DEGs upregulated in tumor cells relative to their expression in juxta-tumoral cells), meaning those functional pathways for which in the tumor APC showed enrichment, relative to their presence in the corresponding juxta-tumoral APC. Pathway-enrichment analysis identified the pathways 'actomyosin structure organization' and 'proteinaceous extracellular matrix' in pDCs; 'receptor protein tyrosine kinase signaling' in CD14⁺ DCs; and 'kinetochore' in MonoMacs (Fig. 4a). The major molecules driving the enriched pathways included the growth factor CTGF in pDCs, AGR2 in CD14⁺ DCs, and the mitotic checkpoint BUB1 in MonoMac (Fig. 4b). Because

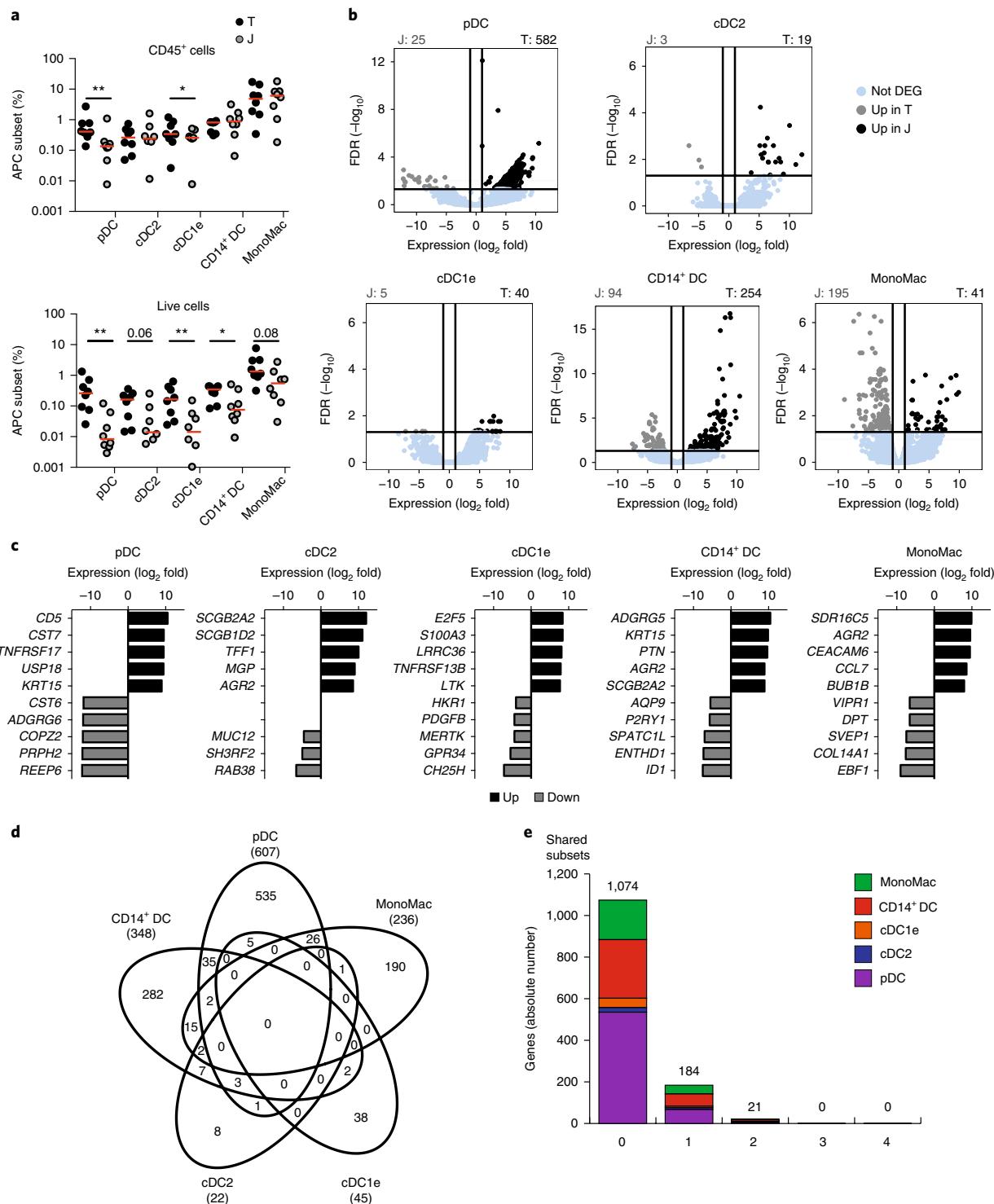


Fig. 3 | 'Tumor-emergent genes' from innate APC are subset specific. **a**, Frequency of APC subsets (horizontal axis) among CD45⁺ cells (top) or total live cells (bottom) in tumor samples (T) and juxta-tumoral samples (J) (key) from patients with LBC, assessed by flow cytometry. Each symbol represents an individual donor ($n=8$; samples paired by donor); small red horizontal lines indicate the median. * $P < 0.05$ and ** $P < 0.01$ (two-tailed Wilcoxon-test).

b, Expression of genes in tumor samples relative to their expression in juxta-tumoral samples (horizontal axis), plotted against the FDR ($FDR < 0.05$; vertical axis), for the transcriptome of each APC subset (above plot), showing genes upregulated in the tumor sample (change in expression of over 1-fold (\log_2 value)) or juxta-tumor sample (change in expression of less than -1-fold (\log_2 value)) or unchanged (Not DEG) (key); numbers above plots indicate the number of DEGs in each group. **c**, Expression of the top five DEGs (left margin) upregulated or downregulated (key) in each APC subset in the tumor, presented relative to their expression in the juxta-tumoral sample (\log_2 value). **d**, Quantification of total DEGs in tumor samples versus juxta-tumoral sample, for each subset (in parenthesis along perimeter), and DEGs shared by various subsets (overlapping loops) or all subsets (center). **e**, Quantification of DEGs unique to a specific subset (0) or shared with one, two, three or four other subsets (horizontal axis), for each APC subset (key); numbers above bars indicate total DEGs per shared group. Number of independent matched donors with LBC (**b–e**) (likelihood ratio test from edgeR R software package): $n=3$ (pDC), $n=4$ (cDC2), $n=4$ (cDC1e), $n=3$ (CD14⁺ DCs) and $n=5$ (MonoMac).

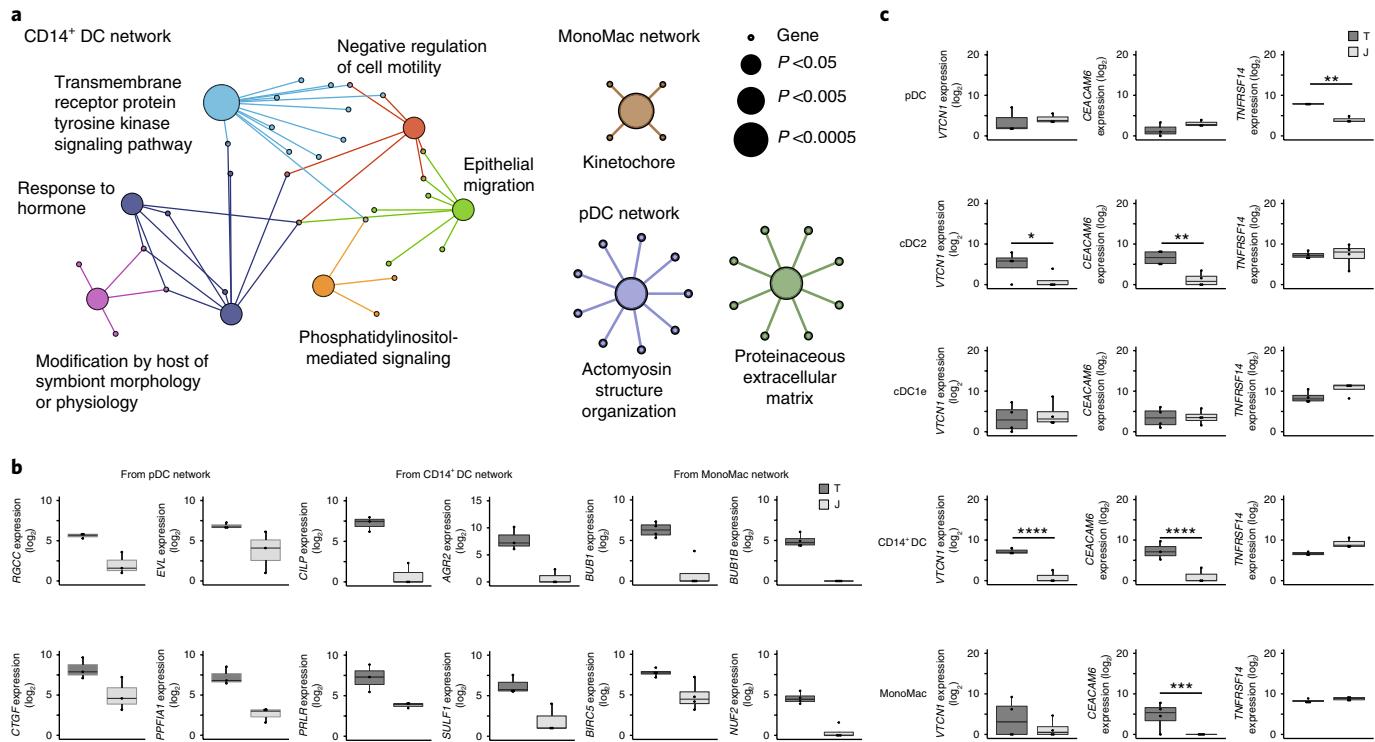


Fig. 4 | Absence of immunological function enrichment in tumor-upregulated genes. **a**, Functional network inference results for the gene signatures ($FDR < 0.05$) of pDCs, CD14⁺ DCs and MonoMacs (above plots) from LBC samples, presented as in Fig. 2d. **b**, Expression of genes encoding molecules in the pathways most significantly enriched (as in **a**) for pDCs, CD14⁺ DCs and MonoMacs (above plots) in tumor or juxta-tumoral samples (key) from LBC, presented as read counts + 1 (\log_2 values); box plots as in Fig. 1e. **c**, Expression of the checkpoint molecule-encoding genes VTCN1, CEACAM6 and TNFRSF14 (entire list, Methods) in tumor and juxta-tumoral samples (key) from LBC, presented as in **b**; only genes expressed differentially in at least one subset are presented. * $P < 0.05$, ** $P < 0.01$, *** $P < 0.005$ and **** $P < 0.0001$ (likelihood ratio test, edgeR). Number of independent matched donors with LBC (**a-c**): $n = 3$ (pDC), $n = 4$ (cDC2), $n = 4$ (cDC1e), $n = 3$ (CD14⁺ DC) and $n = 5$ (MonoMac).

we did not identify any enrichment for immunological function by this unbiased approach, we specifically investigated the expression of immunological checkpoint molecules important in anti-tumor immunity^{28,29}. Out of 19 positive and 15 negative immunological checkpoint molecules, we found that genes encoding the following were expressed differentially in tumor APCs relative to their expression in juxta-tumoral APCs: HVEM (TNFRSF14) in pDCs; B7-H4 (VTCN1) and (CEACAM6) in cDC2s and CD14⁺ DCs; and CEACAM6 in MonoMacs (Fig. 4c). In conclusion, the molecules encoded by ‘tumor-emerging genes’ from LBC APCs were poorly linked to immunological functions.

The transcriptomics profile of tumor APCs depends on the breast-cancer subtype. To evaluate the effect of tumor type on the DC transcriptional profile, we generated the transcriptomes of pDCs, cDC2s and CD14⁺ DCs from four triple-negative breast cancer (TNBC) samples and of cDC1e cells and MonoMacs from four TNBC samples (Supplementary Fig. 3 and Supplementary Table 1). Principal-component analysis of tumor DC transcriptional profiles using the 500 genes with the most-variant expression indicated that DCs clustered by cancer subtype rather than by DC subset (Fig. 5a), suggestive of differential tumor imprinting on DCs. pDCs separated from the other APC subsets in both cancer types (Fig. 5a). To identify the genes upregulated in TNBC relative to their expression in LBC for each DC subset, we performed differential analysis ($FDR < 0.05$, and a change in expression of over onefold (\log_2 values)). MonoMacs had the greatest number of DEGs (2,930), followed by CD14⁺ DCs (2,662s) and pDC (1,434) (Fig. 5b). cDC1e cells (605 DEGs) and cDC2s (521 DEGs) were less affected by tumor

type (Fig. 5b). The majority of DEGs (65% of genes upregulated in TNBC relative to their expression in LBC) were upregulated exclusively in one DC subset (Fig. 5c). Four DEGs (*IFNL1*, *IFNB1*, *ISG20* and *ISG15*), all associated with the interferon pathway, were upregulated in TNBC relative to their expression in LBC (Fig. 5d). These data indicated that two different types of cancer had a major effect on the transcriptomes of the infiltrating DCs and MonoMacs.

TNBC promotes a shared immune system-related signature in DCs. The pDCs had the greatest number of enriched pathways (166) relative to the number of enriched pathways in other APCs (Fig. 6a). MonoMacs, cDC2s and CD14⁺ DCs shared 49%, 36% and 29%, respectively, of their enriched pathways with at least one other subset (Fig. 6a). In contrast, cDC1e cells shared only 6% of their enriched pathways with other subsets (Fig. 6a). These results suggested that pathways for which TNBC APCs showed enrichment were mostly subset specific, indicative of functional specialization for each subset.

We then focused on the pathways for which TNBC APCs commonly showed enrichment. We identified 38 pathways, including those linked to immune system-related functions, that were shared with at least another APC subset (Fig. 6b and Supplementary Fig. 4a). In particular, ‘chemokine activity’, ‘cytokine activity’, ‘cytokine receptor binding’ and ‘IL-10 signaling’ were shared by cDC2s and CD14⁺ DCs (Supplementary Fig. 4a). All DC subsets commonly showed enrichment for type I interferon-related pathways, such as ‘IFN α / β signaling’ and ‘negative regulation of viral life cycle’ (Fig. 6b). From all type I interferon-related pathways, we selected the genes that showed significant enrichment, including *IFNB1*, *ISG15* and

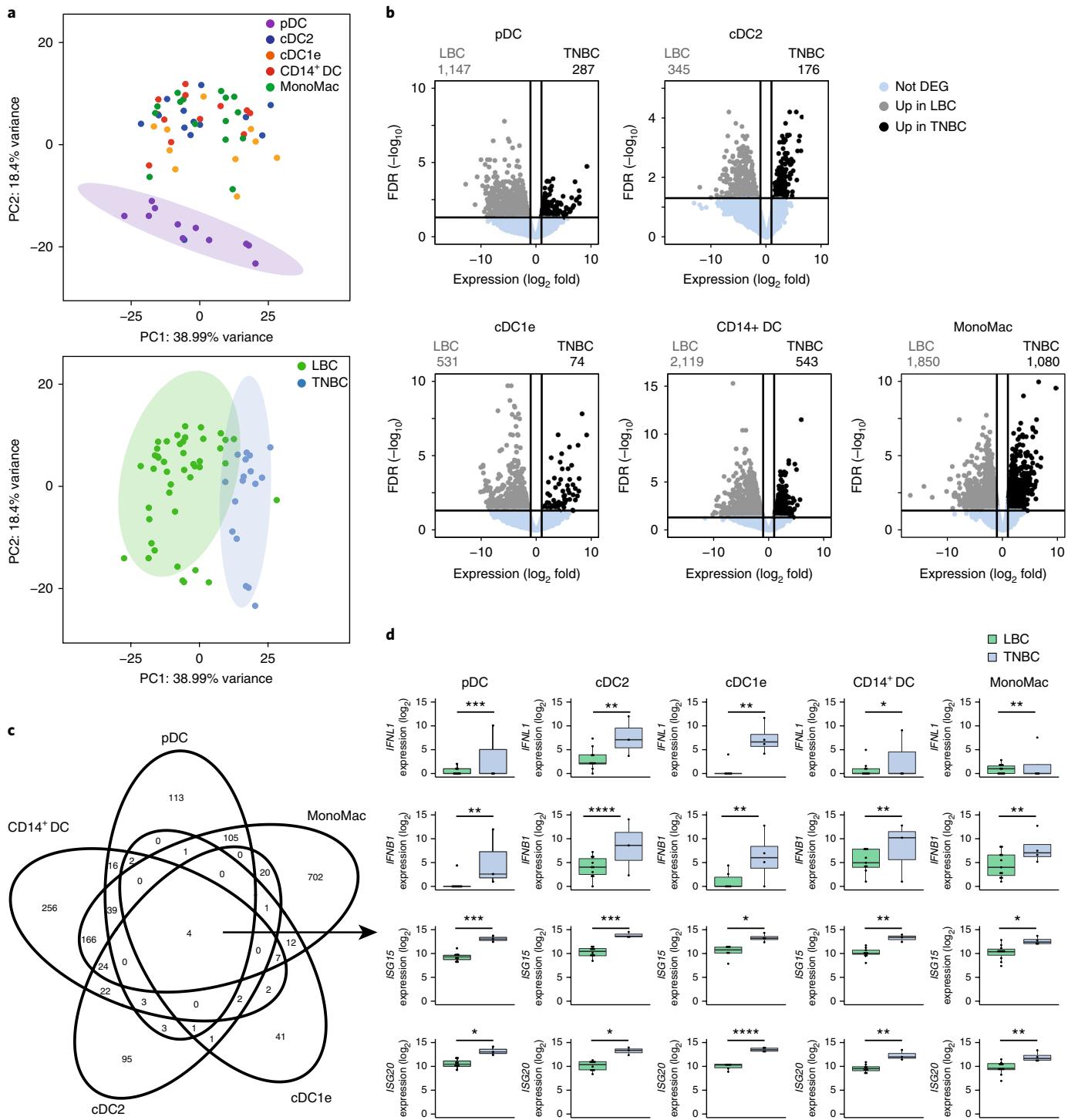


Fig. 5 | Transcriptional profile of innate APC subset is dependent on breast-cancer subtype. **a**, Principal-component analysis showing the clustering of transcriptional profiles (with the 500 most variant genes) of innate APC subsets isolated from LBC or TNBC tumors, with principal components PC1 and PC2 projected and the variance of each (along axes), presented by subset (key; top) or by breast-cancer type (key; bottom). **b**, Expression of genes in TNBC relative to their expression in LBC (horizontal axis), plotted against the FDR ($\text{FDR} < 0.05$; vertical axis), for the transcriptome of each APC subset (above plot), showing genes upregulated in TNBC (change in expression of over 1-fold (\log_2 value)) or LBC (change in expression of less than -1-fold (\log_2 value)) or unchanged (key); numbers above plots indicate the number of DEGs in each group. **c**, Quantification of DEGs shared by various subsets of APCs (overlapping loops) or all subsets (center), among DEGs upregulated in TNBC relative to their expression in LBC. **d**, Expression of the four genes upregulated in all APC subsets in TNBC (center in **c**), assessed in the five APC subsets (above plots) from LBC or TNBC (key), presented as read counts + 1 (\log_2 values); box plots as in Fig. 1e. $*P < 0.05$, $**P < 0.01$ and $***P < 0.005$ (likelihood ratio test, edgeR). Number of samples and donors (**a-d**): $n = 8$ (pDC), $n = 10$ (cDC2), $n = 6$ (cDC1e), $n = 9$ (CD14⁺ DC) and $n = 11$ (MonoMac), from six to ten donors with LBC; and $n = 3$ (pDC), $n = 4$ (cDC2), $n = 3$ (cDC1e), $n = 3$ (CD14⁺ DC) and $n = 4$ (MonoMac), from three to four TNBC donors.

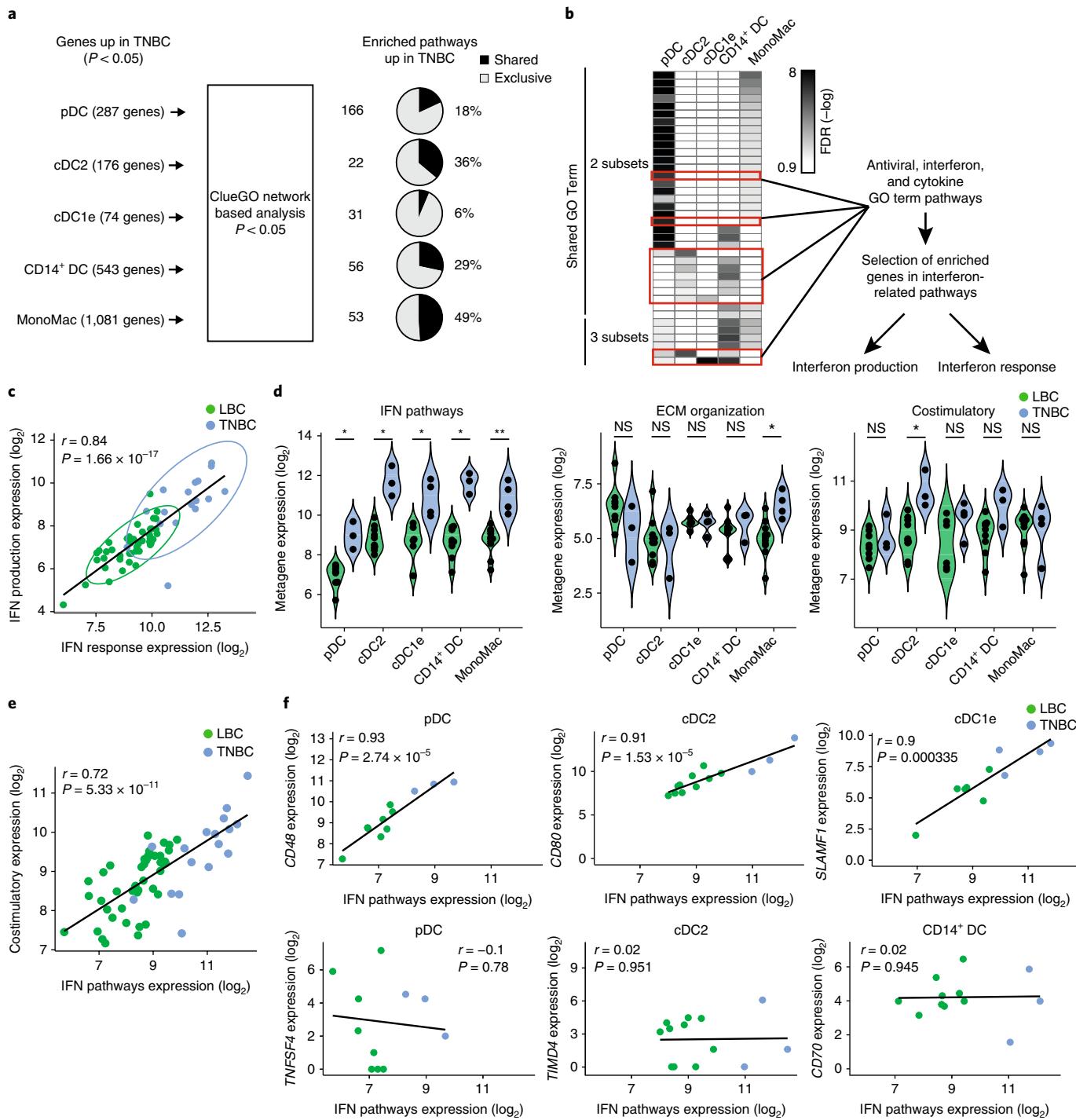


Fig. 6 | The type I interferon pathway is upregulated in all APC subsets in TNBC. **a**, Functional pathways analysis of DEGs upregulated in APCs in TNBC, showing the resultant number of genes in each (left), the number of pathways (middle) and the proportion of shared or specific pathways (key) for each APC subset (right). **b**, Significance (FDR, -log values) for enriched pathways (FDR < 0.05) shared by two or three (left margin) APC subsets (above plots); red outlines indicate the immunological pathways at right (interferon metagenes extracted from significantly enriched pathways categorized as interferon production or interferon response). **c**, Correlation between the expression (\log_2 values) of genes encoding molecules involved in the interferon response (IFN response expression) and those involved in interferon production (IFN production expression), for all APC subsets isolated from LBC or TNBC (key). **d**, Expression of the IFN pathways metagene, the 'ECM organization' pathway metagene and the 'Costimulatory' pathway metagene (above plots) for each APC subset (horizontal axis) from LBC or TNBC (key), presented as a 'violin' plot. NS, not significant ($P > 0.05$); * $P < 0.05$, ** $P < 0.01$ and *** $P < 0.005$ (two-sided Wilcoxon test). **e**, Correlation between expression (\log_2 values) of the IFN pathways metagene and 'Costimulatory' pathway metagene in all APC subsets isolated from LBC or TNBC (key). **f**, Correlation between expression (\log_2 values) of the IFN pathway metagene and the costimulatory molecule-encoding genes *CD48* (*SLAMF2*), *CD80*, *SLAMF1*, *TNFSF4* (*OX40L*), *TIMD4* and *CD70* in all APC subsets (above plots) from LBC or TNBC (key). Numbers in plots (**c,d,f**) indicate the correlation coefficient (r) and P value (Pearson correlation test). Number of samples: $n=8$ (pDC), $n=10$ (cDC2), $n=6$ (cDC1e), $n=9$ (CD14⁺ DC) and $n=11$ (MonoMAC) for LBS, and $n=3$ (pDC), $n=3$ (cDC2), $n=4$ (cDC1e), $n=3$ (CD14⁺ DC) and $n=4$ (MonoMAC) for TNBC.

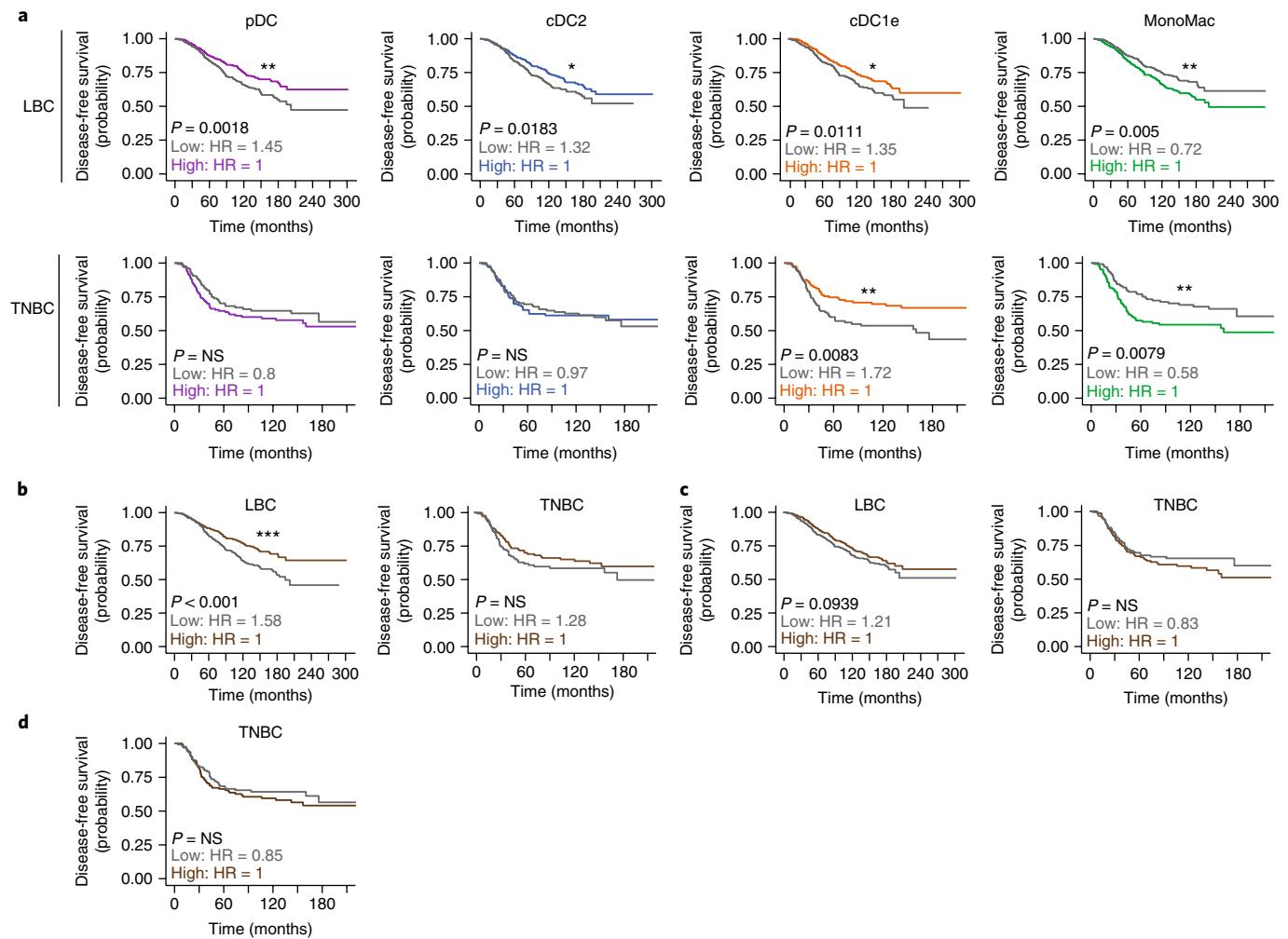


Fig. 7 | Subset-specific signatures are linked to distinct disease-free survival depending on the subset and breast cancer type. **a**, Probability of disease-free survival over time for patients with LBC (top row) or TNBC (bottom) (Kaplan-Meier plots), associated with a high or low z-score (line color) for the signature specific to each APC subset (above plots): line colors indicate the hazard ratio (bottom left in each plot) associated with a low z-score (Low: HR) or high z-score (High: HR); P values (bottom left in each plot), log-rank test. **b,c**, Probability of disease-free survival over time for patients with LBC (left) or TNBC (right), associated with the z-scores for a published CD103⁺ DC gene signature⁴⁴ (**b**) or a published pDC signature⁴ (**c**), presented as in **a**. **d**, Probability of disease-free survival over time in patients with TNBC, associated with z-scores for the APC ‘common IFN’ signature, presented as in **a**. *P < 0.05, **P < 0.005 and ***P < 0.001 (log-rank test). Data are from the METABRIC public dataset: n=1,043 donors (LBC) and n=259 donors (TNBC).

ISG20, and classified them as distinct ‘metagenes’ according to the contribution of the molecules encoded to either interferon production or the interferon response (Supplementary Fig. 4b). Because the amount of expression of each metagene was strongly correlated across all TNBC samples (Fig. 6c), we pooled them into a single ‘IFN pathways’ metagene, which had higher expression in all TNBC APCs than in LBC APCs (Fig. 6d). As a control, the ‘ECM organization pathway’ metagene (Supplementary Fig. 4c) had significantly higher expression only in TNBC MonoMacs (Fig. 6d). We also analyzed the expression of a ‘costimulatory’ metagene (Supplementary Fig. 4d) that had significantly higher expression in TNBC than in LBC only for cDC2s, not for other APCs (Fig. 6d) and whose expression was highly correlated with that of the ‘IFN pathways’ metagene (Fig. 6e). When analyzing the dependence of individual checkpoint molecule-encoding genes with the ‘IFN pathways’ metagene, we found that the expression of genes encoding molecules such as SLAMF2 (CD48) in pDCs, CD80 in cDC2s and SLAMF1 in cDC1e cells was highly correlated with expression of the ‘IFN pathways’ metagene (Fig. 6f). In contrast, the expression of TNFSF4 in pDCs, TIMD4 in cDC2s and of CD70 in CD14⁺ DCs

was not correlated with expression of the ‘IFN pathways’ metagene (Fig. 6f). This revealed two groups of checkpoint molecule-encoding genes that were associated differentially with the ‘IFN pathways’ metagene (Supplementary Fig. 4e). Thus, the transcriptomes of APCs in TNBC differed substantially from those of APCs in LBC, with a common ‘IFN pathways’ metagene being upregulated in all TNBC APCs, indicative of a specific contribution of TNBC to the reprogramming of APCs.

Subset-specific signatures of tumor APCs can be used to predict the survival of patients with breast cancer. To assess whether the APC subset-specific signatures might have a prognostic effect, we took advantage of the publicly available dataset from whole-breast-cancer transcriptome METABRIC, which includes patient-survival clinical annotation⁴³. Because of the differences in the APC transcriptional profiles, we investigated LBC and TNBC datasets separately. We calculated a z-score for each APC subset-specific signature⁴⁴ (Supplementary Fig. 5a). We found that high z-scores for pDCs, cDC2s and cDC1e cells were significantly predictive of disease-free survival for patients with LBC (Fig. 7a). On the contrary,

a high *z*-score for MonoMacs was linked to a bad prognosis for patients with LBC or TNBC (Fig. 7a). A high *z*-score for cDC1e cells was linked to a good prognosis for patients with TNBC, with a greater significance than that for LBC (Fig. 7a). The *z*-scores for pDCs and cDC2s had no prognostic value in TNBC (Fig. 7a). This suggested that the various signatures might have a different clinical effect according to DC subset and breast-cancer type.

A CD103⁺ DC gene signature has been reported to correlate with a good prognosis in several tumor types, including breast cancer⁴⁴. Using the METABRIC dataset, we found that the *z*-score for the CD103⁺ DC gene signature had a significant effect on the survival of patients with LBC but not on the prognosis of patients with TNBC (Fig. 7b). We then assessed the prognostic value of the blood pDC signature⁷. The score for the blood pDC signature had no significant effect on survival outcome for patients with LBC or TNBC (Fig. 7c). Hence, prognostic significance was most efficiently reached in a given tumor through the use of DC signatures generated from the same tumor type. Finally, no prognostic value associated with the signature of common DEGs associated with the interferon pathway was found for patients with TNBC (Fig. 7d), which showed that subset-specific signatures harbored more prognostic information than did a shared signature.

We then determined whether subset-specific signatures could be independently associated with survival when integrated with the Nottingham prognostic index, a reference clinical score that determines survival⁴⁵. We observed that all significant scores in univariate analysis were kept in the multivariate analysis for pDCs, cDC2s, cDC1e cells and MonoMacs in LBC and for cDC1e cells and MonoMacs in TNBC (Table 1), which indicated that subset-specific APC signatures in LBC and TNBC were independent prognostic factors associated with disease-free survival. These results demonstrated the relevance of generating subset- and breast cancer type-specific signatures in predicting clinical outcome.

Discussion

Here we used DC-specific markers to identify resident DC populations (cDC2s, cDC1s and pDCs), MonoMacs and subsets that shared many features with previously described inflammatory DCs (CD14⁺ DCs)^{2,21} to provide broad and systematic coverage of the APC subsets that have been identified in two types of breast cancer (LBC and TNBC).

Our analysis revealed that pDCs were the most distinct APC subtype, as reported before in various tissues and species^{5–7,10,46,47}. We propose that part of this pDC-specific signature is determined by ontogeny, as supported by various genes, such as *CLEC4C*, *GZMB* and *TCF4*, identified in the pDC signatures independently of tissue type^{6,7,10}. Other pDC signature genes, such as those encoding the basal membrane laminins *LAMA4*, *LAMB1* and *LAMC1*, not previously associated with a pDC-specific signature^{6,7}, might be attributed to tissue imprinting or to a combined effect of ontogeny and tissue-driven factors. In contrast to pDCs, CD14⁺ DCs and cDC2s had very close similarity with other subsets. Comparative analyses of DC subsets across multiple studies might identify conserved, ontogeny-determined signatures, rather than more plastic and environment-driven transcriptional modifications.

Among high-throughput studies of tumor-infiltrating APCs in the mouse^{48–50}, only two compared tumor tissue with non-tumor tissue^{48,50}, but such studies focused on a single APC population, such as CD11b⁺ DCs⁵¹ or macrophages^{48,50}, and did not systematically compare diverse APCs in the context of their adaptation to a tumor context. Here, by systematically comparing the transcriptome of each APC subset in tumor tissue with that in uninvolved juxta-tumoral tissue, we identified emergent features in tumor-infiltrating APCs relative to those in non-tumor-tissue APCs. This ‘imprinting’ was different for distinct APC subsets, both qualitatively and quantitatively, which indicated that in breast cancer, there is no unique

signature that can be attributed to tissue imprinting, as has been suggested for other anatomical sites^{6–8}. We propose that the effect of the tissue microenvironment on innate immune cells should be considered and interpreted in close interaction with subset-specific molecular features.

cDC1s have been proposed to be the main APC subset that drives antitumor responses in mouse tumor models in a type I interferon-dependent manner^{44,52,53}. In our study, cDC1e cells expressing genes encoding XCR1 and CLEC9A, as well as other cDC-specific markers, did not have higher expression of genes encoding molecules related to DC activation or antigen presentation than that of the other APC signatures in either LBC or TNBC. Moreover, all human APC transcriptomes from TNBC, not only cDC1e cells, showed enrichment for genes encoding molecules involved in the interferon response and interferon production; this indicated that, at least in human breast cancer, all DCs are able to upregulate an interferon signature. Further experiments are needed to determine whether cDC1s are key to antitumor immune responses in humans.

Tumors have been segregated on the basis of their infiltration by immune cells: poor versus substantial (‘cold tumors’ versus ‘hot tumors’)⁵¹. The first category is characterized by poor infiltration by T cells and an increase in angiogenic and extracellular matrix factors^{54,55}. The second category has greater infiltration by T cells and increased expression of chemokines and type I interferons^{52,54,55}. Both tumor types are associated with distinct mechanisms for escaping the immune system^{51,54,55}. The breast-cancer subtypes we investigated here, TNBC and LBC, have substantial infiltration and poor infiltration, respectively, by immune cells⁵⁶. LBC DCs, and in particular LBC pDCs, showed enrichment for the ‘vascular wound healing’ and ‘extracellular matrix’ pathways, whereas TNBC DC subsets showed enrichment for immunological signatures, including interferon pathways. Hence, our findings have identified DCs as another level for the immune system-based stratification of tumors. This could aid in studies of the differential contributions of DC subsets to the mechanisms used by different tumor types to escape the immune system.

TNBC is a rare and aggressive breast-cancer subtype⁵⁷. Clinical trials using checkpoint blockers in TNBC are ongoing, with promising results^{14,58}. Hence, there is considerable interest in precisely characterizing the immune-system compartment in such patients. Here we have provided detailed analysis of APC subsets in TNBC. In particular, signatures specific to cDC1e cells, but not those specific to pDCs or cDC2s, were predictive of survival in TNBC, in contrast to results obtained for LBC. Hence, our data can be exploited to identify TNBC-specific prognostic signatures, as well as to identify promising targets to better direct therapies targeting immunological checkpoints.

Overall, our study has provided detailed and comprehensive molecular profiling of tumor-infiltrating DC subsets and MonoMacs in human cancer, which might serve as a reference dataset to increase biological knowledge of DCs in the context of disease. Our findings have shed light on the rules that dictate DC diversity and adaptation to complex microenvironments, such as in cancer, through transcriptional reprogramming. Our data will help to delineate the individual contributions of DC subsets to anti-tumor immunity and should provide a valuable resource for the identification of potential targets and biomarkers to better direct cancer immunotherapies.

Methods

Methods, including statements of data availability and any associated accession codes and references, are available at <https://doi.org/10.1038/s41590-018-0145-8>.

Received: 30 March 2017; Accepted: 11 May 2018;
Published online: 16 July 2018

References

1. Banchereau, J. & Steinman, R. M. Dendritic cells and the control of immunity. *Nature* **392**, 245–252 (1998).
2. Collin, M., McGovern, N. & Haniffa, M. Human dendritic cell subsets. *Immunity* **140**, 22–30 (2013).
3. Mildner, A. & Jung, S. Development and function of dendritic cell subsets. *Immunity* **40**, 642–656 (2014).
4. Shay, T. & Kang, J. Immunological Genome Project and systems immunology. *Trends Immunol.* **34**, 602–609 (2013).
5. Miller, J. C. et al. Deciphering the transcriptional network of the dendritic cell lineage. *Nat. Immunol.* **13**, 888–899 (2012).
6. Heidkamp, G. F. et al. Human lymphoid organ dendritic cell identity is predominantly dictated by ontogeny, not tissue microenvironment. *Sci. Immunol.* **1**, eaai7677 (2016).
7. Haniffa, M. et al. Human tissues contain CD141hi cross-presenting dendritic cells with functional homology to mouse CD103+ nonlymphoid dendritic cells. *Immunity* **37**, 60–73 (2012).
8. Watchmaker, P. B. et al. Comparative transcriptional and functional profiling defines conserved programs of intestinal DC differentiation in humans and mice. *Nat. Immunol.* **15**, 98–108 (2014).
9. Liu, Y. J. IPC: professional type 1 interferon-producing cells and plasmacytoid dendritic cell precursors. *Annu. Rev. Immunol.* **23**, 275–306 (2005).
10. Lindstedt, M., Lundberg, K. & Borrebaeck, C. A. Gene family clustering identifies functionally associated subsets of human *in vivo* blood and tonsillar dendritic cells. *J. Immunol.* **175**, 4839–4846 (2005).
11. Mora, J. R. et al. Selective imprinting of gut-homing T cells by Peyer's patch dendritic cells. *Nature* **424**, 88–93 (2003).
12. Huang, Q. et al. The plasticity of dendritic cell responses to pathogens and their components. *Science* **294**, 870–875 (2001).
13. Pulendran, B., Palucka, K. & Banchereau, J. Sensing pathogens and tuning immune responses. *Science* **293**, 253–256 (2001).
14. Stagg, J. & Allard, B. Immunotherapeutic approaches in triple-negative breast cancer: latest research and clinical prospects. *Ther. Adv. Med. Oncol.* **5**, 169–181 (2013).
15. Liu, Y. J. Dendritic cell subsets and lineages, and their functions in innate and adaptive immunity. *Cell* **106**, 259–262 (2001).
16. Dalod, M., Chelbi, R., Malissen, B. & Lawrence, T. Dendritic cell maturation: functional specialization through signaling specificity and transcriptional programming. *EMBO J.* **33**, 1104–1116 (2014).
17. Soumelis, V., Patarini, L., Michea, P. & Cappuccio, A. Systems approaches to unravel innate immune cell diversity, environmental plasticity and functional specialization. *Curr. Opin. Immunol.* **32**, 42–47 (2015).
18. Segura, E. & Amigorena, S. Inflammatory dendritic cells in mice and humans. *Trends Immunol.* **34**, 440–445 (2013).
19. Wollenberg, A., Haberstok, J., Teichmann, B., Wen, S. P. & Bieber, T. Demonstration of the low-affinity IgE receptor FcεRII/CD23 in psoriatic epidermis: inflammatory dendritic epidermal cells (IDEC) but not Langerhans cells are the relevant CD1a-positive cell population. *Arch. Dermatol. Res.* **290**, 517–521 (1998).
20. Zaba, L. C., Krueger, J. G. & Lowes, M. A. Resident and “inflammatory” dendritic cells in human skin. *J. Invest. Dermatol.* **129**, 302–308 (2009).
21. Segura, E. et al. Human inflammatory dendritic cells induce Th17 cell differentiation. *Immunity* **38**, 336–348 (2013).
22. Dunn, G. P., Bruce, A. T., Ikeda, H., Old, L. J. & Schreiber, R. D. Cancer immunoediting: from immunosurveillance to tumor escape. *Nat. Immunol.* **3**, 991–998 (2002).
23. Bell, D. et al. In breast carcinoma tissue, immature dendritic cells reside within the tumor, whereas mature dendritic cells are located in peritumoral areas. *J. Exp. Med.* **190**, 1417–1426 (1999).
24. DeNardo, D. G. & Coussens, L. M. Inflammation and breast cancer. Balancing immune response: crosstalk between adaptive and innate immune cells during breast cancer progression. *Breast Cancer Res.* **9**, 212 (2007).
25. Ghiringhelli, F. et al. CD4+CD25+ regulatory T cells inhibit natural killer cell functions in a transforming growth factor-β-dependent manner. *J. Exp. Med.* **202**, 1075–1085 (2005).
26. Faget, J. et al. ICOS-ligand expression on plasmacytoid dendritic cells supports breast cancer progression by promoting the accumulation of immunosuppressive CD4+ T cells. *Cancer Res.* **72**, 6130–6141 (2012).
27. Ghirelli, C. et al. Breast cancer cell-derived GM-CSF licenses regulatory Th2 induction by plasmacytoid predendritic cells in aggressive disease subtypes. *Cancer Res.* **75**, 2775–2787 (2015).
28. Topalian, S. L., Drake, C. G. & Pardoll, D. M. Immune checkpoint blockade: a common denominator approach to cancer therapy. *Cancer Cell* **27**, 450–461 (2015).
29. Chen, D. S. & Mellman, I. Oncology meets immunology: the cancer-immunity cycle. *Immunity* **39**, 1–10 (2013).
30. Angel, C. E. et al. Cutting edge: CD1a+ antigen-presenting cells in human dermis respond rapidly to CCR7 ligands. *J. Immunol.* **176**, 5730–5734 (2006).
31. Bakdash, G. et al. Expansion of a BDCA1+CD14+ myeloid cell population in melanoma patients may attenuate the efficacy of dendritic cell vaccines. *Cancer Res.* **76**, 4332–4346 (2016).
32. McGovern, N. et al. Human dermal CD14+ cells are a transient population of monocyte-derived macrophages. *Immunity* **41**, 465–477 (2014).
33. Bronte, V. et al. Recommendations for myeloid-derived suppressor cell nomenclature and characterization standards. *Nat. Commun.* **7**, 12150 (2016).
34. Guilliams, M. et al. Unsupervised high-dimensional analysis aligns dendritic cells across tissues and species. *Immunity* **45**, 669–684 (2016).
35. Taieb, J. et al. A novel dendritic cell subset involved in tumor immunosurveillance. *Nat. Med.* **12**, 214–219 (2006).
36. Caminschi, I. et al. Putative IKDCs are functionally and developmentally similar to natural killer cells, but not to dendritic cells. *J. Exp. Med.* **204**, 2579–2590 (2007).
37. Krasselt, M., Baerwald, C., Wagner, U. & Rossol, M. CD56+ monocytes have a dysregulated cytokine response to lipopolysaccharide and accumulate in rheumatoid arthritis and immunosenescence. *Arthritis Res. Ther.* **15**, R139 (2013).
38. Villani, A. C. et al. Single-cell RNA-seq reveals new types of human blood dendritic cells, monocytes, and progenitors. *Science* **356**, eaah4573 (2017).
39. Grünewald, K. et al. Mammaglobin gene expression: a superior marker of breast cancer cells in peripheral blood in comparison to epidermal-growth-factor receptor and cytokeratin-19. *Lab. Invest.* **80**, 1071–1077 (2000).
40. Han, J. H. et al. Mammaglobin expression in lymph nodes is an important marker of metastatic breast carcinoma. *Arch. Pathol. Lab. Med.* **127**, 1330–1334 (2003).
41. Kowalewska, M., Chechlińska, M., Markowicz, S., Kober, P. & Nowak, R. The relevance of RT-PCR detection of disseminated tumour cells is hampered by the expression of markers regarded as tumour-specific in activated lymphocytes. *Eur. J. Cancer* **42**, 2671–2674 (2006).
42. Novershtern, N., Regev, A. & Friedman, N. Physical module networks: an integrative approach for reconstructing transcription regulation. *Bioinformatics* **27**, i177–i185 (2011).
43. Curtis, C. et al. The genomic and transcriptomic architecture of 2,000 breast tumours reveals novel subgroups. *Nature* **486**, 346–352 (2012).
44. Broz, M. L. et al. Dissecting the tumor myeloid compartment reveals rare activating antigen-presenting cells critical for T cell immunity. *Cancer Cell* **26**, 638–652 (2014).
45. Galea, M. H., Blamey, R. W., Elston, C. E. & Ellis, I. O. The Nottingham Prognostic Index in primary breast cancer. *Breast Cancer Res. Treat.* **22**, 207–219 (1992).
46. Gautier, E. L. et al. Gene-expression profiles and transcriptional regulatory pathways that underlie the identity and diversity of mouse tissue macrophages. *Nat. Immunol.* **13**, 1118–1128 (2012).
47. Robbins, S. H. et al. Novel insights into the relationships between dendritic cell subsets in human and mouse revealed by genome-wide expression profiling. *Genome Biol.* **9**, R17 (2008).
48. Franklin, R. A. et al. The cellular and molecular origin of tumor-associated macrophages. *Science* **344**, 921–925 (2014).
49. Ojalvo, L. S., Whitaker, C. A., Condeelis, J. S. & Pollard, J. W. Gene expression analysis of macrophages that facilitate tumor invasion supports a role for Wnt-signaling in mediating their activity in primary mammary tumors. *J. Immunol.* **184**, 702–712 (2010).
50. Pyfferen, L. et al. The transcriptome of lung tumor-infiltrating dendritic cells reveals a tumor-supporting phenotype and a microRNA signature with negative impact on clinical outcome. *OncolImmunology* **6**, e1253655 (2016).
51. Wargo, J. A., Reddy, S. M., Reuben, A. & Sharma, P. Monitoring immune responses in the tumor microenvironment. *Curr. Opin. Immunol.* **41**, 23–31 (2016).
52. Fuertes, M. B. et al. Host type I IFN signals are required for antitumor CD8+ T cell responses through CD8α+ dendritic cells. *J. Exp. Med.* **208**, 2005–2016 (2011).
53. Salmon, H. et al. Expansion and activation of CD103+ dendritic cell progenitors at the tumor site enhances tumor responses to therapeutic PD-L1 and BRAF inhibition. *Immunity* **44**, 924–938 (2016).
54. Gajewski, T. F., Schreiber, H. & Fu, Y. X. Innate and adaptive immune cells in the tumor microenvironment. *Nat. Immunol.* **14**, 1014–1022 (2013).
55. Spranger, S. & Gajewski, T. F. Tumor-intrinsic oncogene pathways mediating immune avoidance. *OncolImmunology* **5**, e1086862 (2015).
56. Stanton, S. E., Adams, S. & Disis, M. L. Variation in the incidence and magnitude of tumor-infiltrating lymphocytes in breast cancer subtypes: a systematic review. *JAMA Oncol.* **2**, 1354–1360 (2016).
57. Foulkes, W. D., Smith, I. E. & Reis-Filho, J. S. Triple-negative breast cancer. *N. Engl. J. Med.* **363**, 1938–1948 (2010).
58. Bianchini, G., Balko, J. M., Mayer, I. A., Sanders, M. E. & Gianni, L. Triple-negative breast cancer: challenges and opportunities of a heterogeneous disease. *Nat. Rev. Clin. Oncol.* **13**, 674–690 (2016).
59. Carpentier, S. et al. Comparative genomics analysis of mononuclear phagocyte subsets confirms homology between lymphoid tissue-resident and dermal XCR1+ DCs in mouse and human and distinguishes them from Langerhans cells. *J. Immunol. Methods* **432**, 35–49 (2016).

Acknowledgements

We thank the Institut Curie Cytometry Core facility for cell sorting; INSERM U932, particularly C. Laurent and A.S. Hamy-Petit, for bioinformatics advice; and S. Alculumbe and P. Vargas for discussions. F. Noël was supported by a fellowship from the French Ministry of Research. This work was supported by funding from INSERM (BIO2012-02, BIO2014-08, HTE2016), Fondation pour la Recherche Médicale, ANR-10-IDEX-0001-02 PSL* and ANR-11-LABX-0043, European Research Council (IT-DC 281987) and CIC IGR-Curie 1428, INCA EMERG-15-ICR-1, la Ligue contre le cancer (labellisation EL2016.LNCC/VaS). High-throughput sequencing was performed by the ICGEx NGS platform of the Institut Curie supported by grants ANR-10-EQPX-03 (Equipex) and ANR-10-INBS-09-08 (France Génomique Consortium), InCa from ANR ('Investissements d'Avenir' program), by the Canceropole Ile-de-France and by the SiRIC-Curie program (SiRIC Grant 'INCa-DGOS- 4654').

Author contributions

P.M. designed and performed experiments, analyzed results and wrote the manuscript; F.N. performed bioinformatics analyses and wrote the manuscript; E.Z., U.C. and C.G.

analyzed results; P.S. and O.A. performed experiments; A.S.-D. and M.G-D. contributed to project management; A.V.-S. contributed to clinical project management and pathology review and provided clinical samples; F.R. contributed to clinical project management; S.A. and E.S. provided strategic advice and revised the manuscript; and V.S. designed experiments, supervised the research and wrote the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information is available for this paper at <https://doi.org/10.1038/s41590-018-0145-8>.

Reprints and permissions information is available at www.nature.com/reprints.

Correspondence and requests for materials should be addressed to V.S.

Publisher's note: Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Methods

Human samples and patient characteristics. Fresh samples of tumor tissues and juxta-tumoral tissues (lacking malignant tumor cells) of untreated patients with breast cancer were obtained from Hôpital de l’Institut Curie (Paris) in accordance with Institut Curie ethical guidelines. LBC and TNBC types were included in the study according to the hormonal receptor status. Patient characteristics are summarized in Supplementary Table 2.

Single-cell suspensions from human samples. Tumor and juxta-tumoral tissue were cut into small pieces and digested in CO₂-independent medium (Gibco) containing 5% FBS (HyClone). 2 mg/ml collagenase I (C0130, Sigma), 2 mg/ml hyaluronidase (H3506, Sigma) and 25 µg/ml DNase (Roche) by three rounds of 15 min of incubation with agitation at 37 °C. The samples were filtered on a 40-µm cell strainer (Fischer Scientific) and were diluted in PBS 1× (Gibco) supplemented with 1% de-complemented human serum (BioWest) and EDTA 2 mM (Gibco). After centrifugation, cells were resuspended in the same medium and were counted before being assessed by flow cytometry or sorted.

Antibodies and cell sorting. For phenotypical characterization, single-cell suspensions were stained with antibodies to the following human molecules: CD3-Alexa700 (557943; clone: UCHT1), CD19-Alexa700 (557921; clone: HIB19), CD56-Alexa700 (557919; clone: B159), CD56-BUV737 (564448; clone: NCAM16.2), CD163-BV786 (741003; clone: GHI/61), CD11c-PECy5 (551077; clone: B-ly6), CD11c-PE-CF594 (562393; clone: B-ly6), CD123-BV650 (563405; clone: 7G3), HLA-DR-BUV395 (564040; clone: G46-6), and CD45 allophycocyanin-Cy7 (557833; clone: 2D1), all from BD; CD14-Qdot605 (Q10013; clone: TuK4) from Life Technologies; CD14-BV605 (301833; clone: M5E2), CD16-BV510 (302047 clone: 3G8), CD123-PE-Cy7 (306010; clone: 6h6), CD1c-PE (331506; clone: L161) and HLA-DR BV711 (307643; clone: L243), all from Biolegend; CD1c-PerCP-eFluor710 (46-0015-42; clone: L161) and FceR1-allophycocyanin (17-5899-42; clone: AER-37), all from eBioscience; AXL-AlexaFluor488 (FAB154G; clone: 108724) and CD32B-allophycocyanin (FAB1330A; clone: 190723), both from R&D; and CD141-PE (130-098-841; clone: AD5-14H12) from Miltenyi Biotec. For DC sorting, we used the following antibodies instead of the corresponding marker: CD45-BV570 (304033; clone: HI30), from Biolegend; CD14-FITC (555527; clone: 10.1), from BD; and HLA-DR-allophycocyanin-eFluor780 (47-9956-42; clone: LN3), from eBioscience. Single-cell suspensions of tumor-digested cells were sorted in a BD FACSAria III upgrade using the purity mode, a 100-µm nozzle loop, and at low pressure (20 psi). DC subsets were sorted in Eppendorf tubes containing RPMI plus 5% FBS (HyClone) for morphological analysis. Once the morphology for each subset was confirmed, and because of the low number of tumor-infiltrating APC, we directly sorted tumor APCs in TCL buffer (Qiagen) supplemented with 1% β-mercaptoethanol (SIGMA) for RNA-seq experiments.

Morphological analysis. Sorted cells were subjected to cytocentrifuge and were stained with May–Grunwald–Giems stain. Images were obtained with a ProgRes SpeedXT core 5 Microscope Camera (JENOPTIK) on a Leica DM 4000 B microscope.

RNA-seq. The general RNA-seq workflow is summarized in Supplementary Fig. 1. In brief, RNA from sorted cells (>100 cells) was extracted by using a Single Cell RNA Purification Kit (Norgen Biotech), including on-column DNase digestion (Qiagen), as described by the manufacturer’s protocol. RNA integrity was confirmed with an RNA 6000 Pico Kit (Agilent Technologies) in BioAnalyzer. cDNA was generated with SMARTer Ultra Low input RNA for Illumina Sequencing-HV (Clontech), following manufacturer’s protocol; 14 cycles were used to amplify cDNA. The quantity of cDNA and quality of cDNA were assessed with Qubit dsDNA high sensitivity (Thermofisher) and an Agilent Bioanalyzer using nanochip (Agilent Technologies), respectively. Multiplexed pair-end libraries 50 nt in length were obtained using Nextera XT kit (Clontech). Sequencing was performed in the same batch in Illumina HiSeq 2500 using an average depth of 15 million reads; 50-nt-length reads per samples were obtained. Library sequencing and quality control of the sequencing were performed by the NGS facility at Institut Curie.

RNA-seq data pre-processing. Reads were mapped to the human genome reference (hg19/GRCh37) using TopHat2 software version 2.0.6⁶⁰. Gene expression values were quantified as read counts using HTSeq-count⁶¹. We filtered out genes with fewer than five read counts in at least 25% of samples and normalized the raw data using RUVg method (RUVSeq R package)⁶². This method identifies technical noise based on negative control genes that should be affected by unwanted variations but not affected by biological effects of interest. We selected the 5,000 less-variant genes as negative-control genes. From the 82 samples sequenced, only two were excluded from this study, corresponding to tumor and juxta-tumoral pDC. These samples had low expression of pDC-specific markers and high expression of macrophage markers.

For exploratory analyses, we performed principal-component analysis (PCA) of the 500 most-variant genes, based on inter-quartile range method (IQR) (EMA R package)⁶³, of APC transcriptomes from LBC and TNBC tumor samples. Data were log₂-transformed, centered and scaled. PCA was performed using the FactoMineR

R package. The z-score of log₂-transformed gene expression, scaled by gene, were presented in a heat-map color.

Gene set-enrichment analysis. We selected APC specific gene sets from literature⁵⁹ and performed enrichment analysis on our dataset selected LBC T samples. To do so, we used the BubbleMap module of the BubbleGUM software which perform GSEA with multiple-testing correction⁶⁴.

Statistical analysis. Significant differences in the frequency of APCs among total live cells or CD45⁺ cells were performed using ANOVA, followed by a post-hoc test. For paired samples in the tumor-versus-juxta-tumor comparison of APC, we performed a Wilcoxon test by using the GraphPad Prism 6.0.

To generate subset-specific signature of APC for each condition, we performed one-way ANOVA differential analysis test on the log₂ expression data of the five APCs. We kept only the genes differentially expressed by at least two subsets ($P < 0.05$). We then performed a Tukey post-hoc test to select genes exclusively upregulated in one subset relative to their expression in all the other subsets ($P < 0.05$). Those upregulated genes were defined as the subset-specific signature.

To identify genes whose expression varied between tumor tissues and juxta-tumoral tissues, for each APC separately, we performed pairwise comparison of gene-expression-matched samples using the generalized linear model (GLM) likelihood ratio test of EdgeR R package⁶⁵. Only DEGs with an FDR of <0.05 and a change in expression of over onefold (log₂ values) were considered ‘differentially expressed’. The same analysis was applied to find genes expressed differentially TNBC samples relative to their expression in LBC samples for each subset.

Metagene expression was defined as the median expression (log₂ value) of the genes of interest in each sample. Differential expression analysis of metagenes was done using the non-paired Wilcoxon test. Correlations were assessed using the Pearson correlation test, with a threshold of $P < 0.05$.

All RNA-seq statistical analyses were performed using R software (Version 3.2.3).

Regulatory network and functional inference. We extracted the gene-expression matrix for each subset, and each comparison. The conditions were as followed: 1, one subset versus all other subsets in LBC; 2, one subset versus all other subsets in TNBC; 3, tumor tissue versus juxta-tumoral tissue for each subset separately in LBC; and 4, TNBC versus LBC, for each subset separately. We then loaded the matrix on Cytoscape software version 3.4.0. One analysis per subset was performed. Network inference was performed using ARACNe application, which is based on mutual information theory^{66,67}. The parameters used in ARACNe were Mutual Information Algorithm Type: Variable Bandwidth. We used a transcription factor (TF) list for Hub/TF Definition from the dataset Fantom⁶⁸. The mutual information threshold was 0.5. We next used the ClueGO Application⁶⁹ to determine pathway enrichment in each network. Public datasets only from ‘Experimental evidence’ of Gene Ontology (GO) – Biological process-GOA, - Cellular Component-GOA, - ImmuneSystemProcess-GOA, - Molecular Function-GOA, (updated date: 15.01.2017), InterPro dB: Protein Domains (updated date: 03.11.2015), Reactome (updated date: 20.01.2017), and WikiPathways (updated date: 20.01.2017) were used. The Go Term Fusion option was selected. Only pathways with a Benjamini–Hochberg (BH) adjusted P value below 0.05 were kept.

Checkpoint expression analysis. The presence of the following immunological checkpoints was analyzed among DEGs in tumor samples versus juxta-tumoral samples, for each subset. Positive checkpoint genes included those encoding CD40, CD70, CD80, CD83, TNFSF9 (4-1BB), ICOSL, SEMA4A, TIMD4, C10orf54 (VISTA or B7-H5), TNFRSF13C (BAFFR), TNFSF13 (APRIL), TNFSF13 (HVEML), CD84, CD48, TNSF4 (OX40L) and PVR (CD155). Negative checkpoint genes included those encoding CD274 (PD-L1), CD276 (B7-H3), PDCD1LG2 (PD-L2), BTLA, LGALS1, LGALS3, LGALS9, CD279 (PD1), CEACAM6 and CD209 (DC-SIGN).

Clinical outcome of subset-specific signature score in public breast cancer dataset. METABRIC is a public dataset⁴³ of transcriptomics data of breast tumor samples with clinical data associated. From this dataset, we selected samples from LBC ($n = 1,043$) and TNBC ($n = 259$) according to the expression of the receptors ER, PR and HER2. To study the clinical outcome of patients, we considered those with the label ‘d-d.s.’ and ‘a’ in the ‘last follow up status’ variable. Similar to a published report⁴⁴, we calculated a z-score ratio of upregulated APC subset-specific signatures to downregulated APC subset-specific signatures that we generated from our breast cancer RNA-seq data, as follows:

$$\log_2 \text{Ratio} = \log_2 \left(\frac{\text{Signature UP}}{\text{Signature DOWN}} \right)$$

$$z\text{-score ratio} = \frac{\log_2 \text{Ratio} - \bar{\log_2 \text{Ratio}}}{\text{sd}(\log_2 \text{Ratio})}$$

To assess the predictive value of the CD103⁺ DCs reported before⁴⁴, we applied the same z-score, based on the CD103⁺ DC signature, as the ‘signature UP’, and the CD103-DC signature as the ‘signature DOWN’. The CD103⁺ DC and CD103- DC signatures contained 9 genes and 16 genes, respectively⁴⁴.

To assess predictive value of the pDC signature reported before⁴⁵, we applied the same z-score, based on pDC upregulated genes as the ‘signature UP’ (440 genes) and pDC downregulated genes as the ‘signature DOWN’ (524 genes).

To assess predictive value of the interferon signature found in TNBC APCs, we performed a z-score on the log₂ mean expression of *IFNL1*, *IFNB1*, *ISG15* and *ISG20*. We performed univariate Cox analysis to assess the link between subset-specific signatures z-score ratio expression and disease-free survival. We divided the subset-specific z-score expression in two groups: ‘high’ or ‘low’, according to the median value. Kaplan-Meier curves were generated using survminer R package. Multivariate cox analysis was performed to link subset-specific signatures and the clinical prognostic parameter, Nottingham Prognostic Index (NPI)⁴⁵, to disease-free survival.

Reporting Summary. Further information on experimental design is available in the Nature Research Reporting Summary linked to this article.

Data availability. RNA-seq data that support the findings of this study have been deposited in the NCBI Sequence Read Archive (SRA) with the accession code PRJNA380940.

References

60. Kim, D. et al. TopHat2: accurate alignment of transcriptomes in the presence of insertions, deletions and gene fusions. *Genome Biol.* **14**, R36 (2013).
61. Anders, S., Pyl, P. T. & Huber, W. HTSeq—a Python framework to work with high-throughput sequencing data. *Bioinformatics* **31**, 166–169 (2015).
62. Risso, D., Ngai, J., Speed, T. P. & Dudoit, S. Normalization of RNA-seq data using factor analysis of control genes or samples. *Nat. Biotechnol.* **32**, 896–902 (2014).
63. Servant, N. et al. EMA - A R package for Easy Microarray data analysis. *BMC Res. Notes* **3**, 277 (2010).
64. Spinelli, L., Carpentier, S., Montaña Sanchis, F., Dalod, M. & Vu Manh, T. P. BubbleGUM: automatic extraction of phenotype molecular signatures and comprehensive visualization of multiple Gene Set Enrichment Analyses. *BMC Genom.* **16**, 814 (2015).
65. Robinson, M. D., McCarthy, D. J. & Smyth, G. K. edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics* **26**, 139–140 (2010).
66. Margolin, A. A. et al. ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC Bioinforma.* **7**, S7 (2006).
67. Basso, K. et al. Reverse engineering of regulatory networks in human B cells. *Nat. Genet.* **37**, 382–390 (2005).
68. Joshi, A. et al. Technical Advance: Transcription factor, promoter, and enhancer utilization in human myeloid cells. *J. Leukoc. Biol.* **97**, 985–995 (2015).
69. Bindea, G. et al. ClueGO: a Cytoscape plug-in to decipher functionally grouped gene ontology and pathway annotation networks. *Bioinformatics* **25**, 1091–1093 (2009).

2.2 The inconvenience of data of convenience: computational research beyond post-mortem analyses

Chloé-Agathe Azencott, Tero Aittokallio, Sushmita Roy, **DREAM Idea Challenge Consortium**, Thea Norman, Stephen Friend, Gustavo Stolovitzky & Anna Goldenberg

DREAM Idea Challenge Consortium:

Ankit Agrawal, Tero Aittokallio, Chloé-Agathe Azencott, Emmanuel Barillot, Nikolai Bessonov, Deborah Chasman, Urszula Czerwinska, Alireza Fotuhi Siahpirani, Stephen Friend, Anna Goldenberg, Jan Greenberg, Manuel Huber, Samuel Kaski, Christoph Kurz, Marsha Mailick, Michael Merzenich, Nadya Morozova, Arezoo Movaghar, Mor Nahum, Torbjörn E M Nordling, Thea Norman, Robert Penner, Sushmita Roy, Krishnan Saha, Asif Salim, Siamak Sorooshyan, Vassili Soumelis, Alit Stark-Inbar, Audra Sterling, Gustavo Stolovitzky, S S Shiju, Jing Tang, Alen Tosenberger, Thomas Van Vieet, Krister Wennerberg & Andrey Zinovyev

Published in Nature Methods on 29 September 2017

One of burning problems of computational scientists is the fact that the data ideal to verify some hypothesis born from theoretical work or simulations do not exist. Idea Dream Challenge was a call for projects that would describe the ideal data for a proposed model. Winning project would obtain money to gather necessary data. All projects would participate in matching board that aimed to expose interesting theoretical work with experimental scientist that may have or produce necessary data.

I proposed a project, together with my thesis supervised Andrei Zinovyev and Vassili Soumelis. We proposed three independent datasets that could be used to study TME.

1. A single cell data from tumor transcriptomes filling requirements of minimal number of cells of each type to facilitate the statistical analysis.
2. A bulk transcriptome data of systematically co-culture immune-related cells of different types together controlling their proportions with sufficient number of combinations (at least several hundreds) of different cell type proportions in order to study cell-cell interactions. This data would contain 1) individual transcriptomic profiles of pure cell cultures (few tens, containing the replicas) and 2) transcriptomic profiles of controlled mixtures of cell cultures (if possible, containing combinations of many cell types).
3. A benchmark dataset for deconvolution methods: bulk transcriptomic profiles of tumoral samples coupled with carefully quantified proportions of the immune-related cells of different types and the tumoral cellularity .

In the project description we presented as well the ICA model of deconvolution and our preliminary results.

Our project was selected in the first but not the second round of the review process.

All participant of the Idea Dream Challenge co-authored the correspondence to *Nature Methods* as the DREAM Idea Challenge Consortium.

The inconvenience of data of convenience: computational research beyond post-mortem analyses

To the Editor: Over the last two decades researchers have witnessed an explosion in the amount and diversity of data collected in biological and medical studies. These data are often generated without the input of those who will later analyze it. Computational analyses are therefore, in the words of statistician Ronald Fisher, mostly performed ‘post mortem’. We believe that a more efficient scientific process should use computational modeling based on previously acquired data to guide targeted data collection efforts.

We consider systematic data collection and model-driven data collection as distinct efforts. Large-scale systematic data collection efforts, such as TCGA, ENCODE, REMC, GTEx and the Connectivity Map, to name a few, have unquestionably led to important and actionable findings such as identifying treatment targets (<https://cancergenome.nih.gov/researchhighlights/tcgainaction/tcga-data-used-for-loxo101-drug-development>) and gaining insight into gene regulatory processes¹. However, such data could have been even more useful. For example, in our own work on glioblastoma subtype discovery², we could use only 46% of the TCGA samples because of missing measurements, reducing the power of the study. In another example, the fixed concentration levels of small-molecule compounds in the Connectivity Map were suboptimal for some compounds and cell contexts, leading to substantial batch effects³.

DREAM Challenges, which harness the collective skills of computational biologists across the world to solve biological and medical problems using ‘data of convenience’, have illustrated the difficulties in this process^{4–6}. For instance, in a DREAM challenge for predicting response to drugs in patients with rheumatoid arthritis, using the largest available collection of single-nucleotide polymorphism (SNP) data did not improve predictions over those obtained using the clinical predictors⁵. In a toxicogenetic challenge, genome-wide association study (GWAS) data by themselves were not predictive, but the results were markedly better when these were taken together with RNA-seq data, available for only 38% of the patients⁴. Finally, in a DREAM challenge assessing and improving drug sensitivity prediction algorithms, having data from many omics modalities did not provide an advantage over the use of gene expression data alone⁶. We concede that these situations may arise because some computational approaches are just not good enough for the task. However, the fact that none of several dozen independent expert teams were successful in solving the problems using the same data suggests that, instead, more or different kinds of data may be needed. The question then arises: How can one efficiently determine which data we *need to*, rather than *can*, measure to accelerate scientific discovery?

Hypothesis-driven experiments are common in the life sciences but tend to be small in scale. We argue that computational models, capable of generating targeted hypotheses that capture the complexity of biological systems, should be used to guide data collection. This offers the possibility not only of speeding up data collection but also of yielding better biological insights, thanks to the

exploitation of more appropriate data. Recent successes in physics, such as the discovery of gravitational waves and the Higgs boson, illustrate the benefits of model-based experimentation very well. The biomedical field needs such examples of its own.

We firmly believe that computational biologists can contribute productively to model-driven experimental research. Models derived from more classical post-mortem data analysis should now guide the next wave of hypothesis generation, experimental design and data collection. To identify biomedical problems ready to be tackled, we have invited computational biologists from around the world to take part in the Idea DREAM Challenge (<http://tinyurl.com/dreamidea>). Participants were asked to propose biomedical research questions for which computational models have exploited available data to the limit and are ready to guide new data collection efforts to move the field forward. Through peer review and discussions among participants, we selected two winning ideas. We are now matching the winning participants with wet-lab researchers to generate the necessary data.

The first idea addresses the challenge of drug–target interaction mapping. The potential chemical space of drug-like compounds is thought to contain on the order of 10^{20} molecules, making exhaustive exploration infeasible. Furthermore, currently available bioactivity measurements vary greatly between labs and assay types, and hence are not yet sufficient to reliably guide the computational prediction of compound–target relationships at a large scale. One of the winning DREAM ideas proposed a model-guided experimental design and mapping effort to prioritize the most potent target selectivity experiments among the massive search space of compounds and their potential targets. Such targeted experiments, which will be predicted by computational models, are expected to offer a cost-effective alternative to the more systematic exploration efforts, effectively providing higher information content with the same amount of experiments.

The other winning DREAM idea tackles the problem of regulatory network inference, predicting which regulatory proteins control the expression of which target genes. The proposal is to systematically and iteratively collect multi-omic measurements under different genetic and environmental perturbations from both bulk populations and single cells. These data will be collected in a model-guided manner, in which the initial model is a consensus derived from published datasets to avoid duplication of experimental effort and enable maximal discovery. The resulting dataset will serve as a better gold standard to validate computational predictions from existing and new inference methods and will help identify the most informative datasets for regulatory network discovery.

We envision that the Idea DREAM Challenge is just the beginning of many more endeavors in which data analysts and computational biologists can be actively engaged in all stages of the scientific process. Model builders and experimentalists would benefit from working together to design better studies that will accelerate scientific discovery.

COMPETING FINANCIAL INTERESTS

The authors declare no competing financial interests.

Chloé-Agathe Azencott^{1–3}, **Tero Aittokallio**^{4,5}, **Sushmita Roy**^{6,7},
DREAM Idea Challenge Consortium⁸, **Thea Norman**⁹,
Stephen Friend⁹, **Gustavo Stolovitzky**^{10,11} &
Anna Goldenberg^{12,13}

CORRESPONDENCE

¹MINES ParisTech, PSL-Research University, CBIO—Centre for Computational Biology, Fontainebleau, France. ²Institut Curie, Paris, France. ³INSERM U900, Paris, France. ⁴Institute for Molecular Medicine Finland (FIMM), University of Helsinki, Finland. ⁵Department of Mathematics and Statistics, University of Turku, Finland. ⁶Department of Biostatistics & Medical Informatics, University of Wisconsin–Madison, Madison, Wisconsin, USA. ⁷Wisconsin Institute for Discovery, University of Wisconsin–Madison, Madison, Wisconsin, USA. ⁸A list of members and affiliations follows. ⁹Sage Bionetworks, Seattle, Washington, USA. ¹⁰IBM Computational Biology Center, Yorktown Heights, New York, USA. ¹¹Department of Genetics and Genomics Sciences, Icahn School of Medicine at Mount Sinai, New York, New York, USA. ¹²Genetics and Genome Biology, SickKids Research Institute, Toronto, Ontario, Canada. ¹³Department of Computer Science, University of Toronto, Toronto, Ontario, Canada.

DREAM Idea Challenge Consortium:

Ankit Agrawal¹⁴, Tero Aittokallio^{4,5}, Chloé-Agathe Azencott^{1–3}, Emmanuel Barillot¹⁵, Nikolai Bessonov¹⁶, Deborah Chasman⁷, Urszula Czerwinska¹⁵, Alireza Fotuhi Siahpirani¹⁷, Stephen Friend⁹, Anna Goldenberg^{12,13}, Jan Greenberg¹⁸, Manuel Huber¹⁹, Samuel Kaski^{20,21}, Christoph Kurz¹⁹, Marsha Mailick²², Michael Merzenich²³, Nadya Morozova^{24,25}, Arezoo Movaghar^{22,26}, Mor Nahum²³, Torbjörn E M Nordling²⁷, Thea Norman⁹, Robert Penner^{25,28,29}, Sushmita Roy^{6,7}, Krishanu Saha^{7,22,26}, Asif Salim³⁰, Siamak Sorooshyari²³, Vassili Soumelis³¹, Alit Stark-Inbar²³, Audra Sterling^{22,32}, Gustavo Stolovitzky^{10,11}, S S Shiju³⁰, Jing Tang^{4,5}, Alen Tosenberger^{25,33}, Thomas Van Vieet²³, Krister Wennerberg⁴ & Andrey Zinov'yev¹⁵

¹⁴The Institute of Mathematical Sciences, HBNI, CIT Campus, Taramani, Chennai, India. ¹⁵Institut Curie, PSL Research University, Mines ParisTech, Inserm U900, Paris, France. ¹⁶Institute of Problems of Mechanical Engineering, Russian Academy of Sciences, St. Petersburg, Russia. ¹⁷Department of Computer Sciences, University of Wisconsin–Madison, Madison, Wisconsin, USA. ¹⁸Department of Social Work, University of Wisconsin–Madison, Madison, Wisconsin, USA. ¹⁹Institute of Health Economics and Health Care Management, Helmholtz Zentrum München (GmbH)–German Research Center for Environmental Health, Neuherberg, Germany. ²⁰Department of Computer Science, Aalto University, Helsinki, Finland. ²¹Helsinki Institute for Information Technology HIIT, Aalto University, Helsinki, Finland. ²²Waisman Center, University of Wisconsin–Madison, Madison, Wisconsin, USA. ²³Posit Science, San Francisco, California, USA. ²⁴Laboratoire Epigénétique et Cancer, CNRS FRE 3377, CEA Saclay, Gif-sur-Yvette, France. ²⁵Institut des Hautes Etudes Scientifiques (IHES), Bures-sur-Yvette, France. ²⁶Department of Biomedical Engineering, University of Wisconsin–Madison, Madison, Wisconsin, USA. ²⁷Department of Mechanical Engineering, National Cheng Kung University, Tainan, Taiwan. ²⁸Math and Physics Departments, California Institute of Technology, Pasadena, California, USA. ²⁹Centre for the Quantum Geometry of Moduli Spaces, Aarhus University, Aarhus, Denmark. ³⁰Indian Institute of Space Science and Technology, Department of Space, Trivandrum, India. ³¹Institut Curie, PSL Research University, Inserm U932, Paris, France. ³²Department of Communication Sciences and Disorders, University of Wisconsin–Madison, Madison, Wisconsin, USA. ³³Unité de Chronobiologie Théorique, Faculté des Sciences, Université Libre de Bruxelles (ULB), Brussels, Belgium.

1. Alipanahi, B., Delong, A., Weirauch, M.T. & Frey, B.J. *Nat. Biotechnol.* **33**, 831–838 (2015).
2. Wang, B. *et al.* *Nat. Methods* **11**, 333–337 (2014).
3. Kibble, M. *et al.* *Drug Discov. Today* **21**, 1063–1075 (2016).
4. Eduati, F. *et al.* *Nat. Biotechnol.* **33**, 933–940 (2015).
5. Sieberts, S.K. *et al.* *Nat. Commun.* **7**, 12460 (2016).
6. Costello, J.C. *et al.* *Nat. Biotechnol.* **32**, 1202–1212 (2014)

2.3 CV: publications, conferences, courses

URSZULA CZERWINSKA

Paris, France
+33 698210283
urszula.czerwinska@cri-paris.org
[/urszulaczerwinska](https://www.linkedin.com/in/urszulaczerwinska)
[@UlaLaParis](https://twitter.com/UlaLaParis)
Personal webpage
<http://urszulaczerwinska.github.io>

EDUCATION

Doctorate in bio-mathematics (ongoing)

Center for Interdisciplinary Research (CRI), Paris Descartes

2015 **Master of Science**

Interdisciplinary Approaches to Life Sciences (AIV) Center for Interdisciplinary Research (CRI), Paris Diderot
Grades: 16.82/20 **summa cum laude** 1st/12

2013 **Double Bachelor of Science**

Biology-Mathematics in UPMC, Paris VI Grades:
14.04/20 **cum laude**

2013 **Exchange Student**

National University of Singapore, Singapore Grades:
3.45/5.0

2010 **Matura**

French-Polish Bilingual High School, Warsaw, Poland Grades:
94.44%/100% **summa cum laude**

LANGUAGES

Polish	native
English	fluent
French	fluent
Spanish	communicative
Russian	basic

COMPETENCES

Systems biology
Computational biology
Cancer Immunology
Programming
Data Science

RESEARCH EXPERIENCE

- 2015- PhD candidate Institute Curie, Paris Descartes, Center for Interdisciplinary Research
Unsupervised deconvolution of bulk omics profiles: methodology and application to characterize the immune landscape in tumors
Systems biology, systems immunology, complex systems
- Internships
- 2015 Paris Descartes University, Paris metro **microbiome study** (3,5 months)
Microbiology

Pasteur Institute, Systems biology lab. **Biomarker discovery of dengue disease** (4,5 months)
Computational biology
- 2014 iGEM competition. Paris Bettencourt team. **Human skin microbiome engineering** (4 months)
Synthetic biology
- Institut Curie, Computational Systems Biology of Cancer team. **Data-driven layout of biological networks** (5 months)
Systems biology
- 2013 Montpellier University II, Systems Biology and Statistical Physics lab, **Mechanistic model of Syk protein kinase and its downstream pathways** (2 months)
Systems biology
- 2012 CNRS Roscoff, Bioinformatic analysis in Marine Biology group. **Development of a GUI for the raw mass spectrometry data through the plaForm Galaxy.org** (1 month)
Bioinformatics

PUBLICATIONS

Czerwinska, Urszula, et al. "Application of Independent Component Analysis to Tumor Transcriptomes Reveals Specific and Reproducible Immune-Related Signals." *International Conference on Latent Variable Analysis and Signal Separation*. Springer, Cham, 2018.

Azencott, Chloé-Agathe, et al. "The inconvenience of data of convenience: computational research beyond post-mortem analyses." *Nature Methods* 14.10 (2017): 937-938. Collaborators: (...) **Czerwinska U** (...).

Kairov, Ulykbek, Laura Cantini, Alessandro Greco, Askhat Molkenov, **Urszula Czerwinska**, Emmanuel Barillot, and Andrei Zinovyev. "Determining the optimal number of independent components for reproducible transcriptomic data analysis." *BMC genomics* 18, no. 1 (2017): 712.

Naldi, Aurélien, Romain M. Larive, **Urszula Czerwinska**, Serge Urbach, Philippe Montcourrier, Christian Roy, Jérôme Solassol, Gilles Freiss, Peter J. Coopman, and Ovidiu Radulescu. "Reconstruction and signal propagation analysis of the Syk signaling network in breast cancer cells." *PLoS computational biology* 13, no. 3 (2017): e1005432.

Beal, Jacob, et al. "Reproducibility of Fluorescent Expression from Engineered Biological Constructs in *E. coli*." *PloS one* 11.3 (2016): e0150182. (as **iGEM Interlab study contributor**)

Czerwinska , Urszula et al. "DeDaL: Cytoscape 3 app for producing and morphing data-driven and structure-driven network layouts." *BMC systems biology* 9.1 (2015): 1.

Gildas Le Corguillé, Pierre Pericard, **Urszula Czerwinska**, Marion Landi, Franck Giacomoni, Christophe Duperier, Jean-François Martin, Sophie Goulitquer, Estelle Pujos-Guillot and Christophe Caron. A Small Step into Galaxy, a Faster Pace for Metabolomics. Galaxy Community Conference 2013, Oslo (Talk).

CONFERENCES

LVA ICA signal deconvolution and latent variables conference, 2018, Guilford, UK - poster presentation, shout-out talk

RECOMB 2018, conference, Paris, France - poster, volunteering

Data Science Summer School, 2017, l'Ecole Polytechnique, France – poster

Young Researchers in Life Science conference, 2017, Paris, France – talk

ISMB conference, 2017, Prague, Czech Republic, poster

ICSB conference, 2016, Barcelone, Spain, poster

ISMB Conférence, 2016, Orlando, Florida, USA – oral presentation, poster – poster prize, travel grant

Apligoole workshop 2016, Luchon, Fréance, - talk

Conférence Internationale Francophone sur l'Extraction et la Gestion des Connaissances, Atelier de Grand Graphs et Informatique, 2016, Reims, France - talk

BEeSy Conférence, 2015, Grenoble, France – poster prize, talk, travel grant

iGEM convention, 2014, Boston, MA, USA - best application award, Art & Design award, gold medal – as a part of Paris Bettencourt team

Dresden Summer School in Systems, 2013, Dresden, Germany – 2nd award in computational modelling competitor, travel grant

SELECTED ATTENDED COURSES DURING Ph.D. (2015-2018)

GRADUATE SCHOOL COURSES

Public speaking - Scientific Presentations
Scientific writing
Managing scientific collaborations
Typesetting for Scientists
Extended Scientific literacy (7 days)

OTHER

Disruptive Technologies and Public Policy, SciencesPo Paris (54 hours/1 semester)
Python course, HackinScience (7 days)
Systems Immunology, ENS IBENS,
Elevator pitch, Training Unit, Institut Curie
Certificate of Business and Administration, Paris Cité Sorbonne (1 month)
Scientific Integrity, Training Unit, Institut Curie
Big data, Big dive course, Turin, Italy (1 month)

ONLINE

MOOC Machine Learning Standford Coursera by Andrew Ng
MOOC Data Science Essentials by Microsoft EdX

OTHER EXPERIENCE

- 2018** **Teaching** – « Mission enseignement », Pharmacology Faculty of Paris Descartes, Statistics, IT and Mathematics to 2nd and 5th year students
- 2017** **Data Science Club founder** and manager at the Center of Interdisciplinary Research, networking and communication
- 2016** **Blogging** – conference field reports for PLOS Computational Biology Blog
Public speaking 2nd jury award et audience award in a professional pitch at PhD Talent fair
- 2015** **Data Science Ambassador** at Pivigo, Data science hub, London: networking and communication as @UlaLaParis
Start-up co-founder - Eco-Smart Solutions at accelerator L'Open Lab, Paris. Developing probiotic cleaners
- 2014** **General Secretary of an association Open Science School** - Innovative education for high schools
Tutoring in computational Biology - Licence FdV 2nd year
Volunteering: event animator - Pint of Science, Paris, **community manager**: administrator of the Facebook page
Communication and networking: article writing, events and workshops organization at WAX-science, Association for science without stereotypes , Paris, France

Glossary

Biological terms

bulk data - a pooled assay using a weighted average of a bulk cell sample from a particular tissue (i.e., a large population of cells), obscuring cell-to-cell variation

cancer biopsy

cytotoxic

efectors

gene enrichment

gene expression genome immunosuppressive inducers ligands liquid tissue marker gene molecular biology omic data phenotyping receptors solid tissue subtyping tumor purity

Mathematical terms

basis matrix - in cell-type deconvolution, the characteristic expression profiles for each of the cell types to be estimated used in the regression-based deconvolution

bayesian

condition number - a function with respect to an argument measures how much the output value of the function can change for a small change in the input argument. A problem with a low condition number is said to be **well-conditioned**, while a problem with a high condition number is said to be **ill-conditioned**. In linear regression the condition number can be used as a diagnostic for multicollinearity [?].

correlation

covariance

deconvolution - *Math*:the resolution of a convolution function into the functions from which it was formed in order to separate their effects; *Common*: a process of resolving something into its constituent elements or removing complication [?]

diagonal matrix dimension reduction eigenvalue eigenvector matrix matrix factorisation meta-gene metasample monte carlo simulations

multicollinearity

p-value

Post Scriptum: Thesis writing

This Thesis is written in [bookdown](#). I have chosen this form as it can easily compile to *LaTeX*, PDF, MS Word, ebook and html. Optimally, the final manuscript will be also published online in a form of an open source [gitBook](#) and an ebook including interactive figures and maybe even data demos. Another good reason for using [bookdown](#) is its simple syntax of markdown and natural integration of code snippets with .Rmd. It reduces formatting time and give multiple outputs.

The template of for this thesis manuscript was adapted from *LaTeX* template provided by University Paris Descartes.

Citations are stocked in Mendeley Desktop and exported to .bib files automatically.