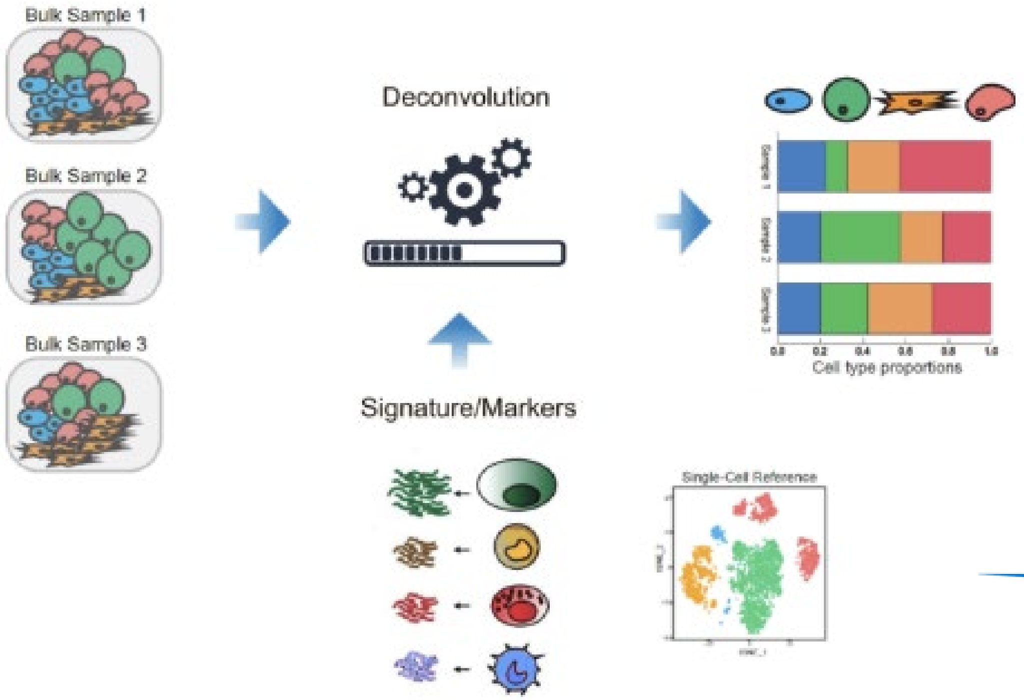
Robust deconvolution of transcriptomic samples using the gene covariance structure

Bastien CHASSAGNOL

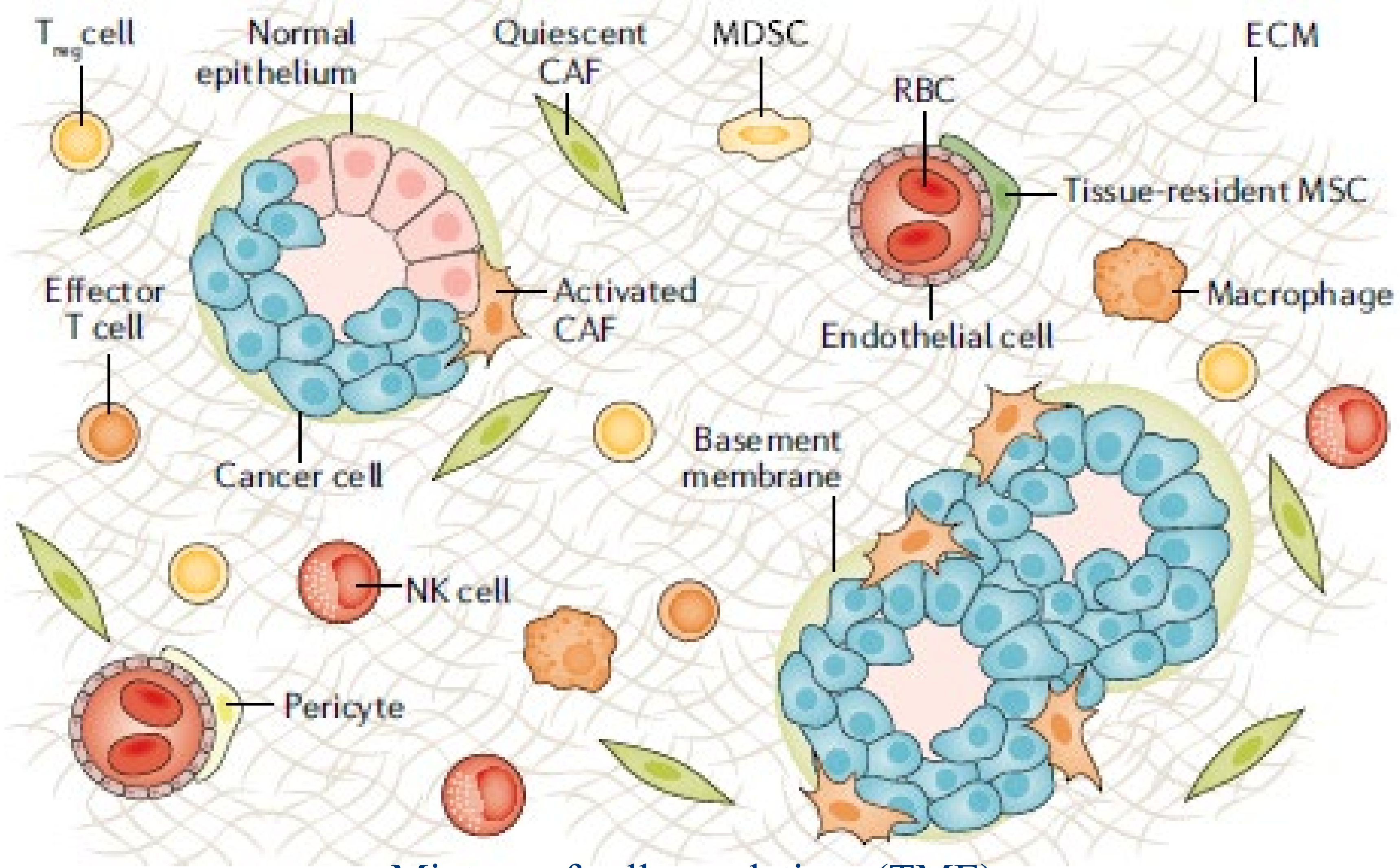
PhD CIFRE financed by Servier, 29/06/2022

Conférence IA et santé

Gregory NUEL, Pierre-Henri WUILLEMIN,
Etienne BECHT and Mickaël GUEDJ



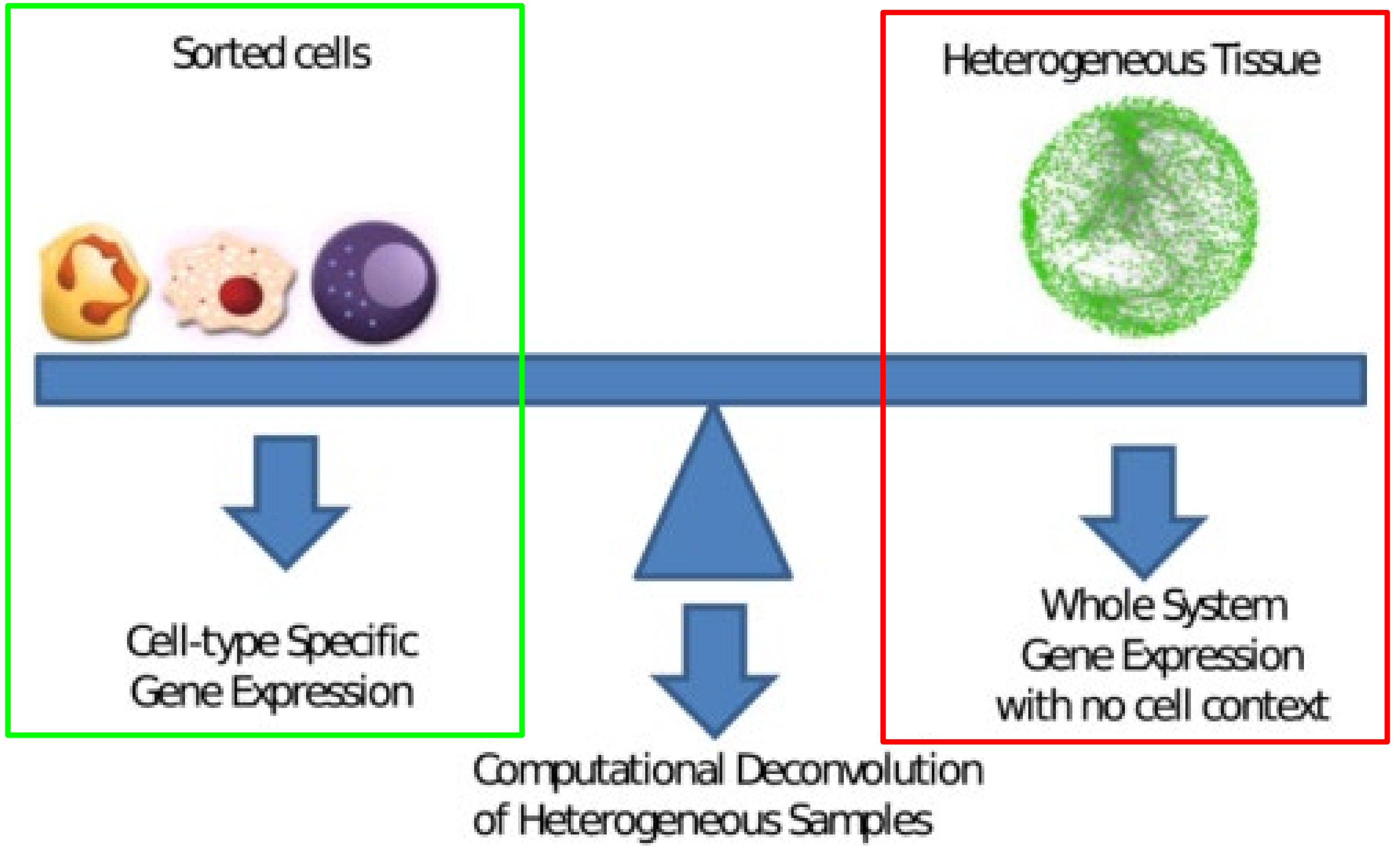
The complexity of the biological medium



Mixture of cell populations (TME)

Finotello and Trajanoski 2018

Physical methods to analyse the biological medium


Sorted cells
Heterogeneous Tissue
Cell-type Specific Gene Expression
Whole System Gene Expression with no cell context
Computational Deconvolution of Heterogeneous Samples

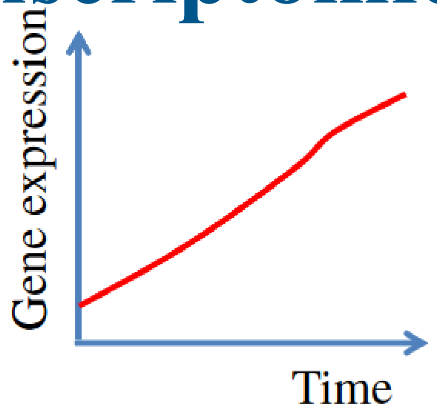
Shen-Orr et al, 2013

Before numerical deconvolution, dilemma between either characterising the individual cell populations (FACS, IHC) or getting a whole transcriptomic(RNASeq, microarray) overview.

Identify the causal transcriptomic driver

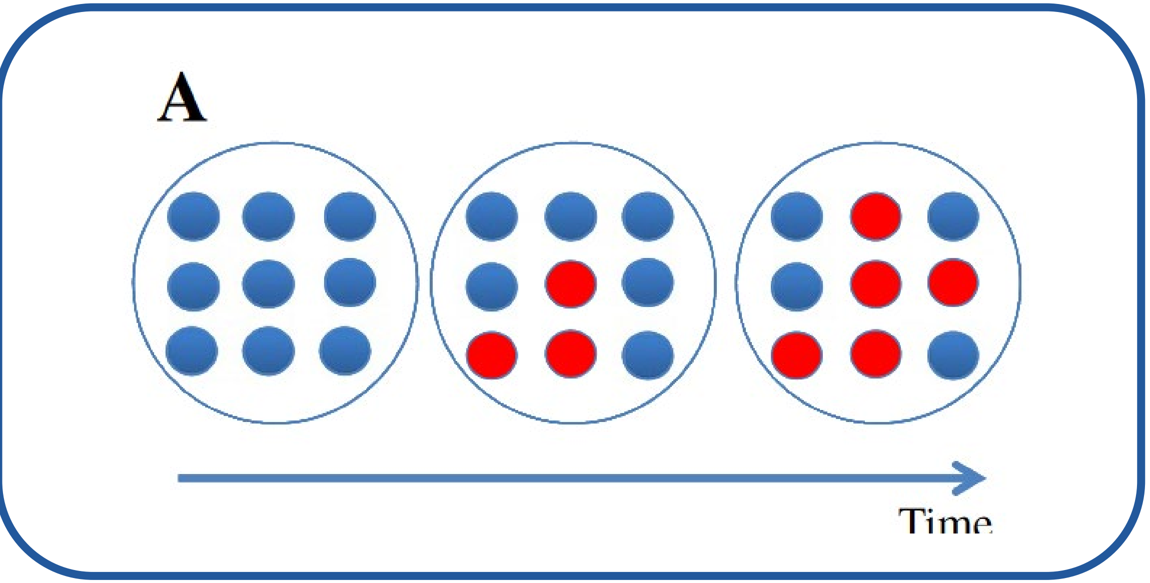
🔵 resting cell population 1

🔴 activated cell population 1

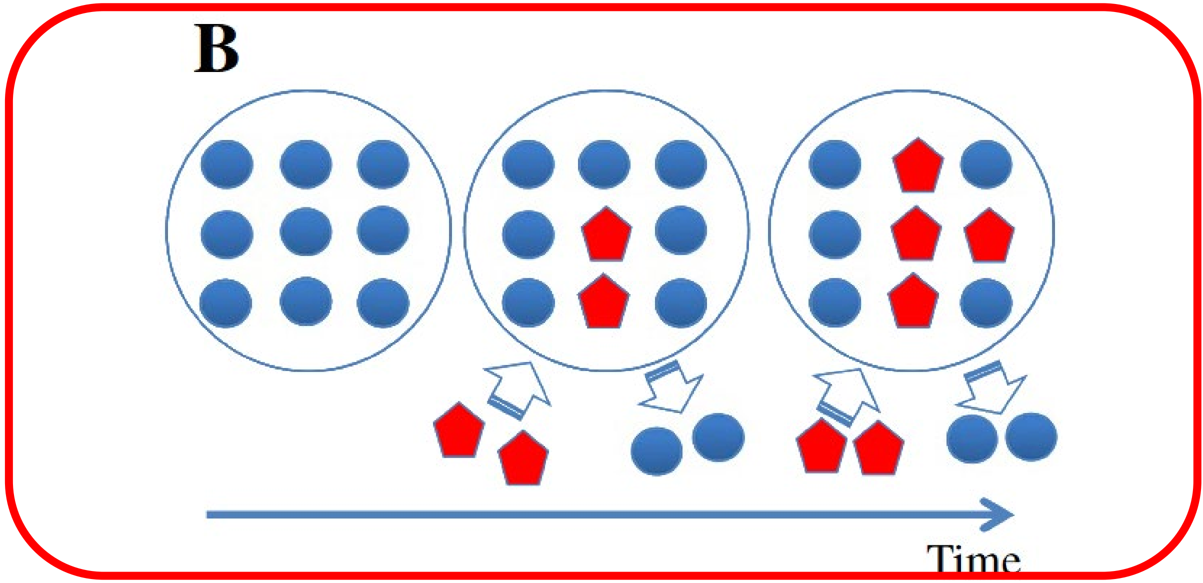


Shoemaker et al. 2012

⬟ cell population 2

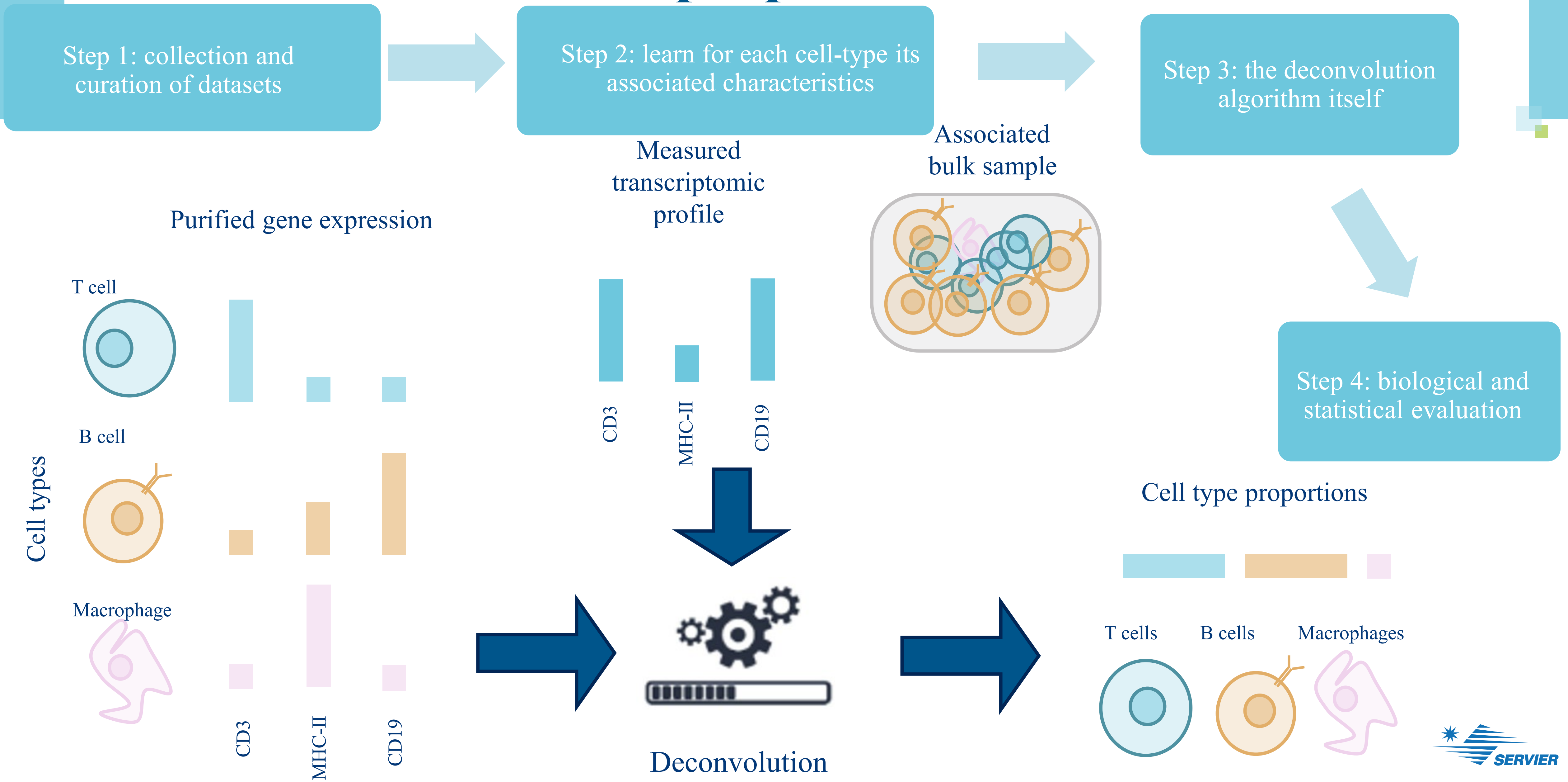


Scenario A: increase of the gene expression is generated by an **activation** of cell population 1



Scenario B: the gene expression increases due to the **infiltration** of a **new** cell population 2

Estimate the cellular proportions



Step 1: collection and curation of datasets

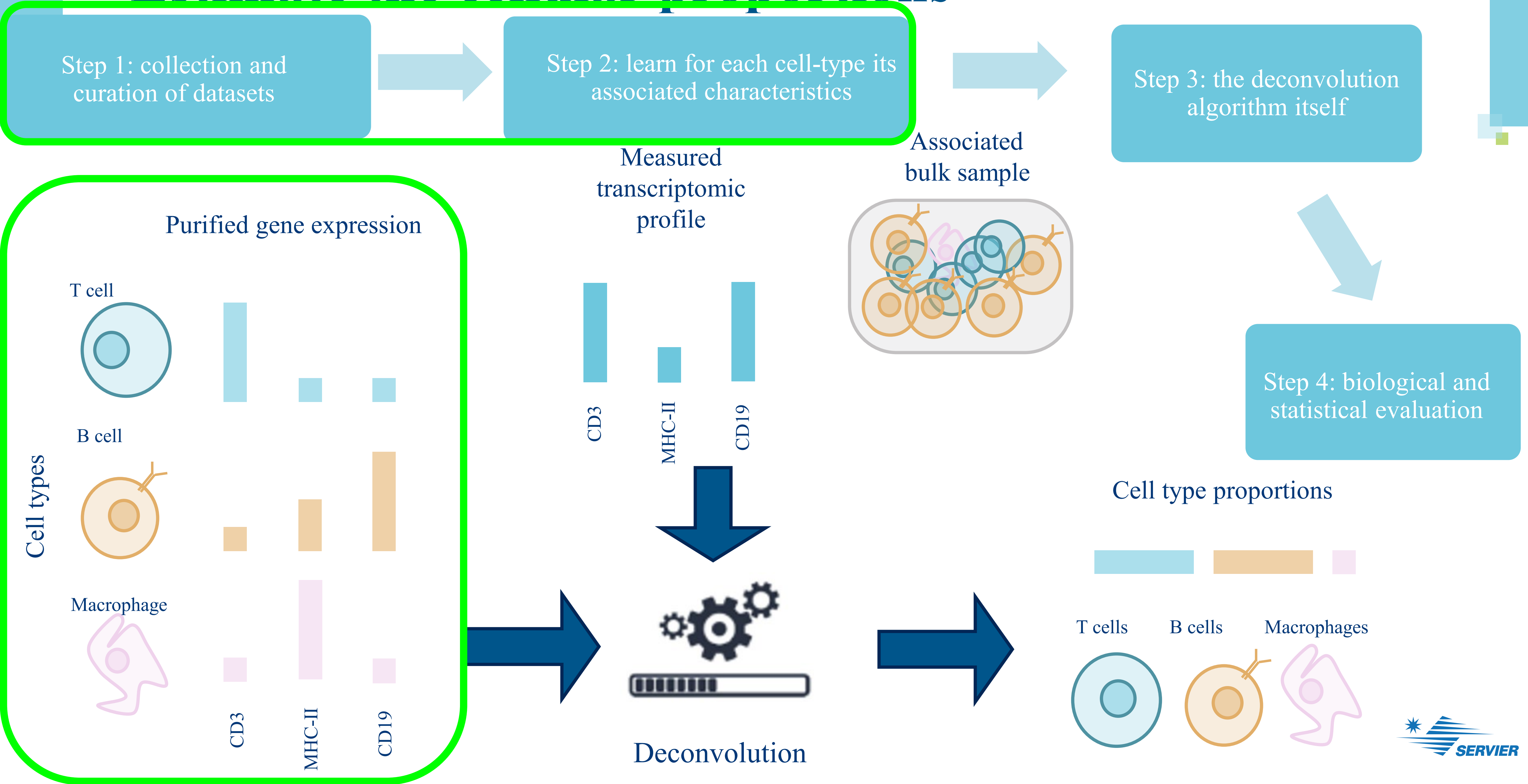
Step 2: learn for each cell-type its associated characteristics

Step 3: the deconvolution algorithm itself

Step 4: biological and statistical evaluation

General principle of cellular deconvolution

Estimate the cellular proportions



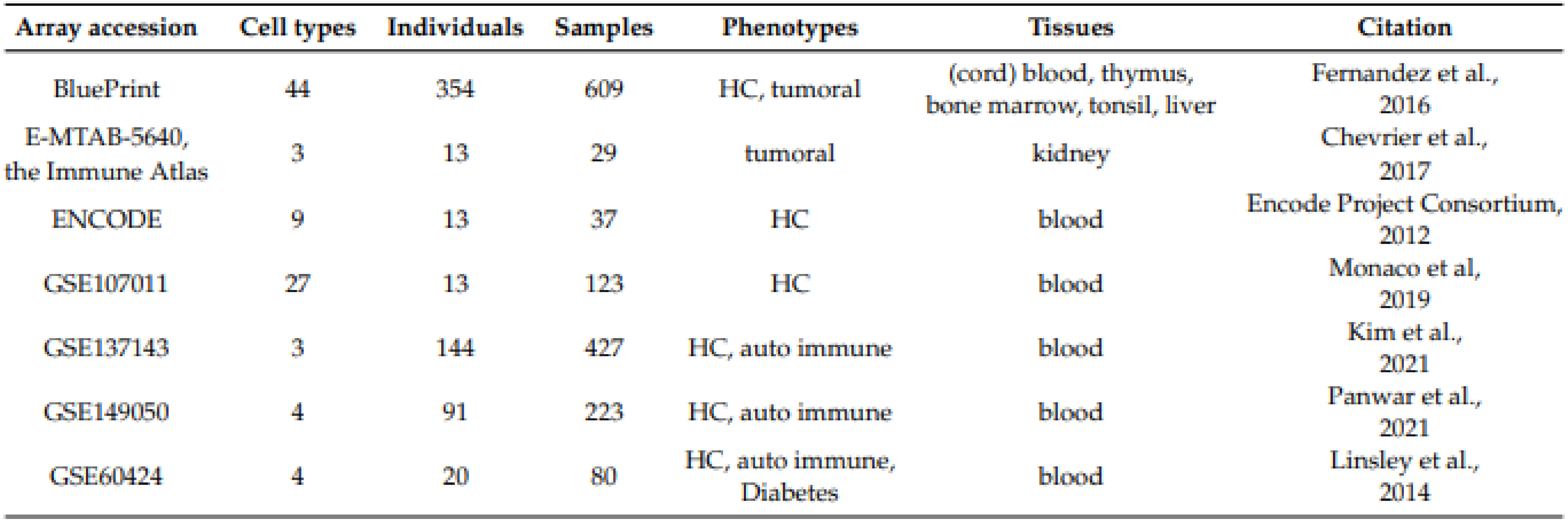
Step 1: collection and curation of datasets

Step 2: learn for each cell-type its associated characteristics

Step 3: the deconvolution algorithm itself

Step 4: biological and statistical evaluation

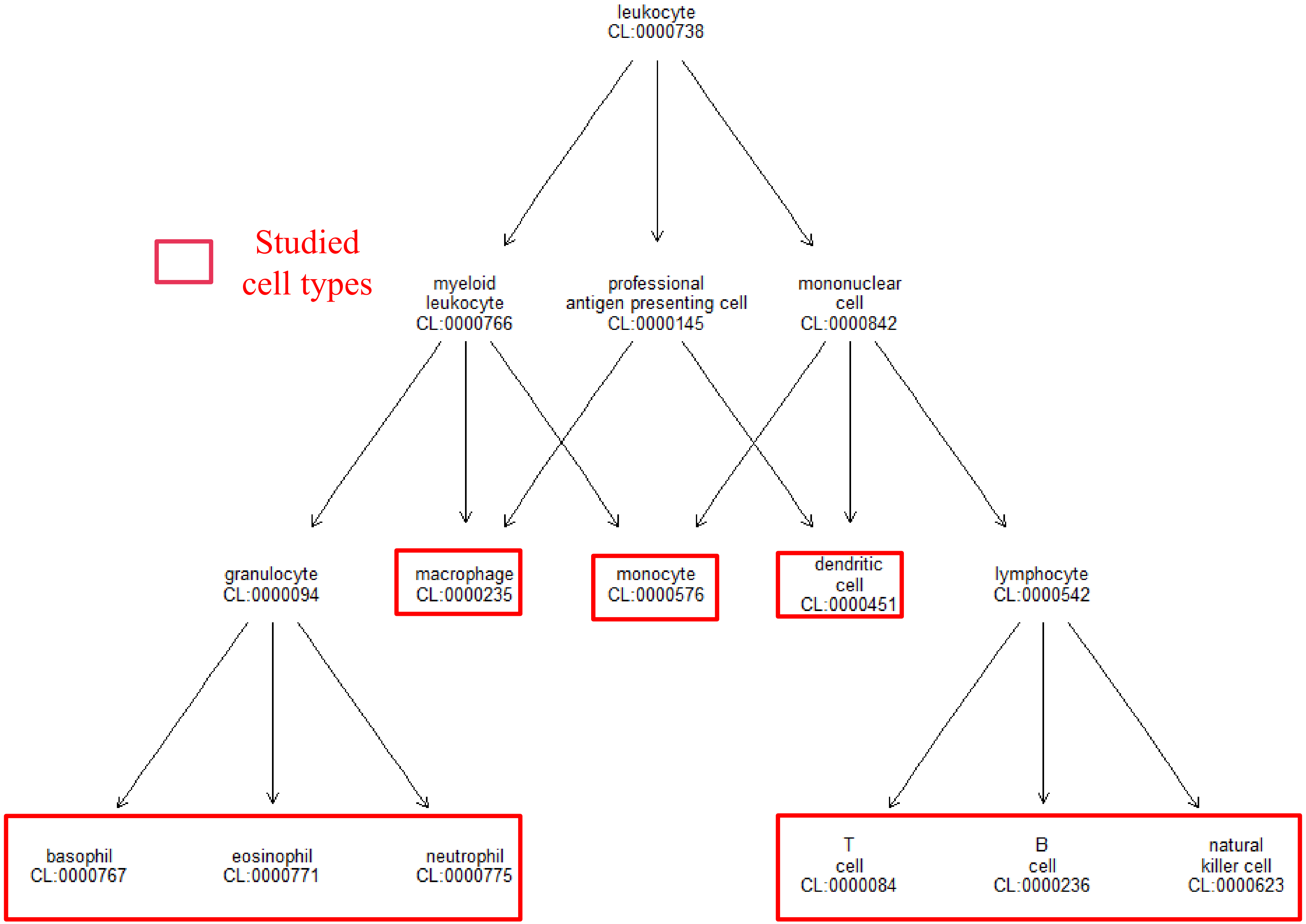
Step 1: selection of the relevant datasets



Array accession	Cell types	Individuals	Samples	Phenotypes	Tissues	Citation
BluePrint	44	354	609	HC, tumoral	(cord) blood, thymus, bone marrow, tonsil, liver	Fernandez et al., 2016
E-MTAB-5640, the Immune Atlas	3	13	29	tumoral	kidney	Chevrier et al., 2017
ENCODE	9	13	37	HC	blood	Encode Project Consortium, 2012
GSE107011	27	13	123	HC	blood	Monaco et al, 2019
GSE137143	3	144	427	HC, auto immune	blood	Kim et al., 2021
GSE149050	4	91	223	HC, auto immune	blood	Panwar et al., 2021
GSE60424	4	20	80	HC, auto immune, Diabetes	blood	Linsley et al., 2014

7 reference RNASeq datasets of purified cell types, covering a large diversity of distinct cell populations (75 unique entities), mostly immune cell types, in 8 distinct tissues (mostly whole blood) and both healthy, tumoral and inflammatory conditions.

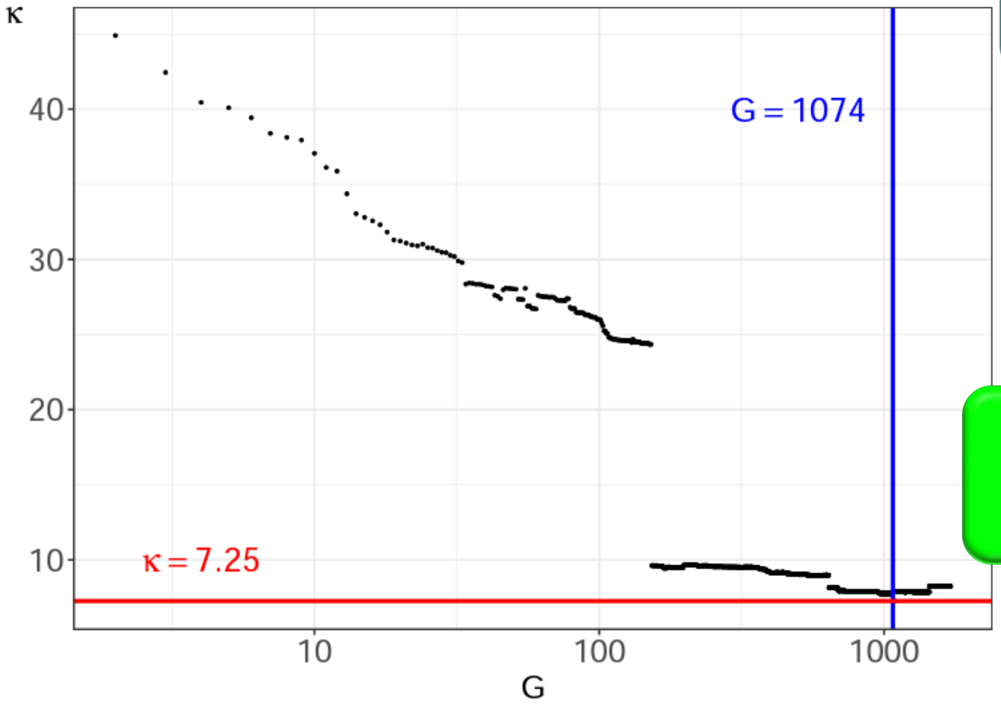
Step 1: selection of the relevant datasets



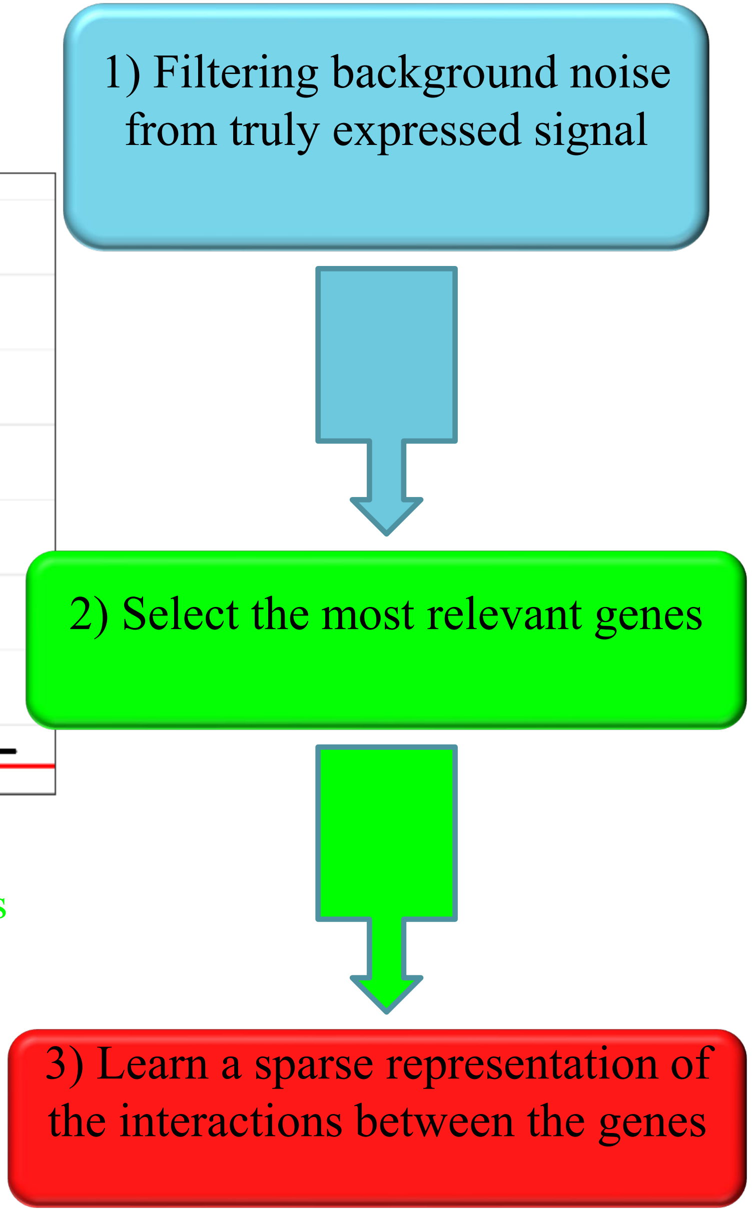
🟥 Studied cell types

Use of *ontoProc* (*Channing, 2022*) to generate automatically the cell ontology from the Human cell atlas.

Step 2: learn the sparse GGM for each cell type



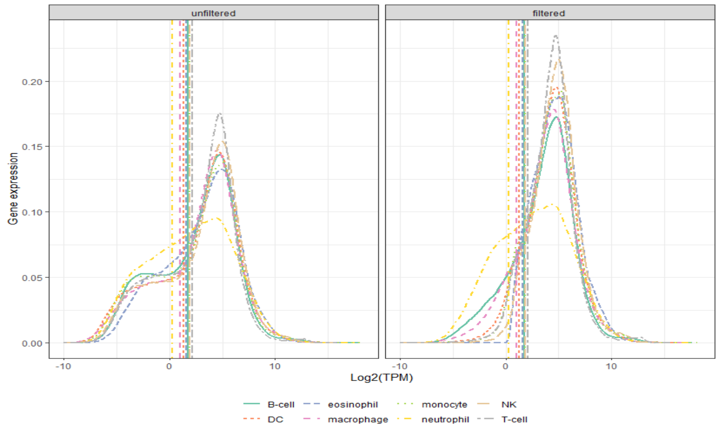
In (*Newman et al, 2015*), selection of the *G* genes associated to the lowest *condition number*.



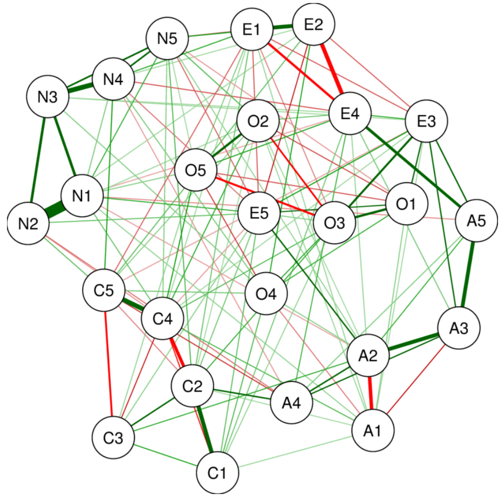
1) Filtering background noise from truly expressed signal

2) Select the most relevant genes

3) Learn a sparse representation of the interactions between the genes



Fitted distributions before and after filtering using zFPKM (*Hart et al, 2013*) process



Nodes represent the genes, and the undirected *edges* the connections between them.

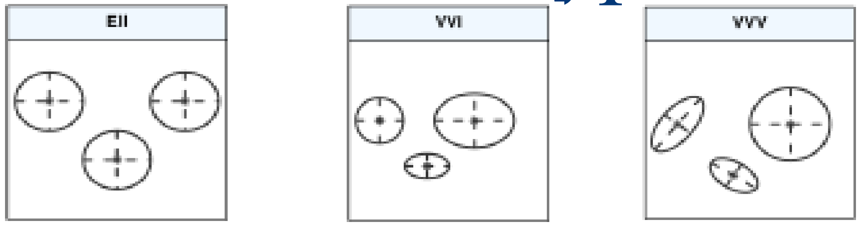
Step 2: learn the sparse GGM for each cell type

Multivariate gaussian distribution

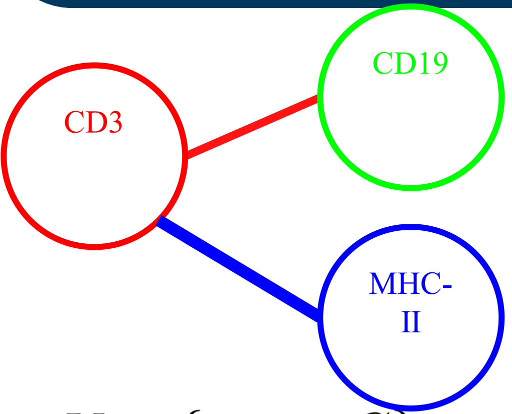
$$\mathbf{X}_{1:G,j} \sim \mathcal{N}_G\left(\boxed{\mu_j}, \boxed{\boldsymbol{\Sigma}_j}\right)$$

$\mu = \mathrm{E}(\mathbf{X})$
Mean vector

$\boldsymbol{\Sigma}_{i,l} = \mathrm{Cov}(X_i, X_l), \forall\, 1 \leq i, l \leq G$
Covariance matrix



Spherical *Diagonal* *Ellipsoidal*



$$\begin{array}{cc} \begin{matrix} \text{CD3} & \text{CD19} & \text{MHC-II} \end{matrix} & \\ \begin{pmatrix} \sigma_1 & 0.5 & 0.8 \\ & \sigma_2 & 0 \\ & & \sigma_3 \end{pmatrix} & \begin{matrix} \text{CD3} \\ \text{CD19} \\ \text{MHC-II} \end{matrix} \end{array}$$

$V = \{1, \dots, G\}$

$E = \{i, l \in V^2, i \neq l\}$

Estimate a sparse covariance structure using gLasso (*Friedman et al, 2008*) algorithm

Precision matrix: the inverse of the covariance matrix

$$\Theta = (\theta_{il}, \quad (i,l) \in \{1, \dots, G\}) = \Sigma^{-1}$$

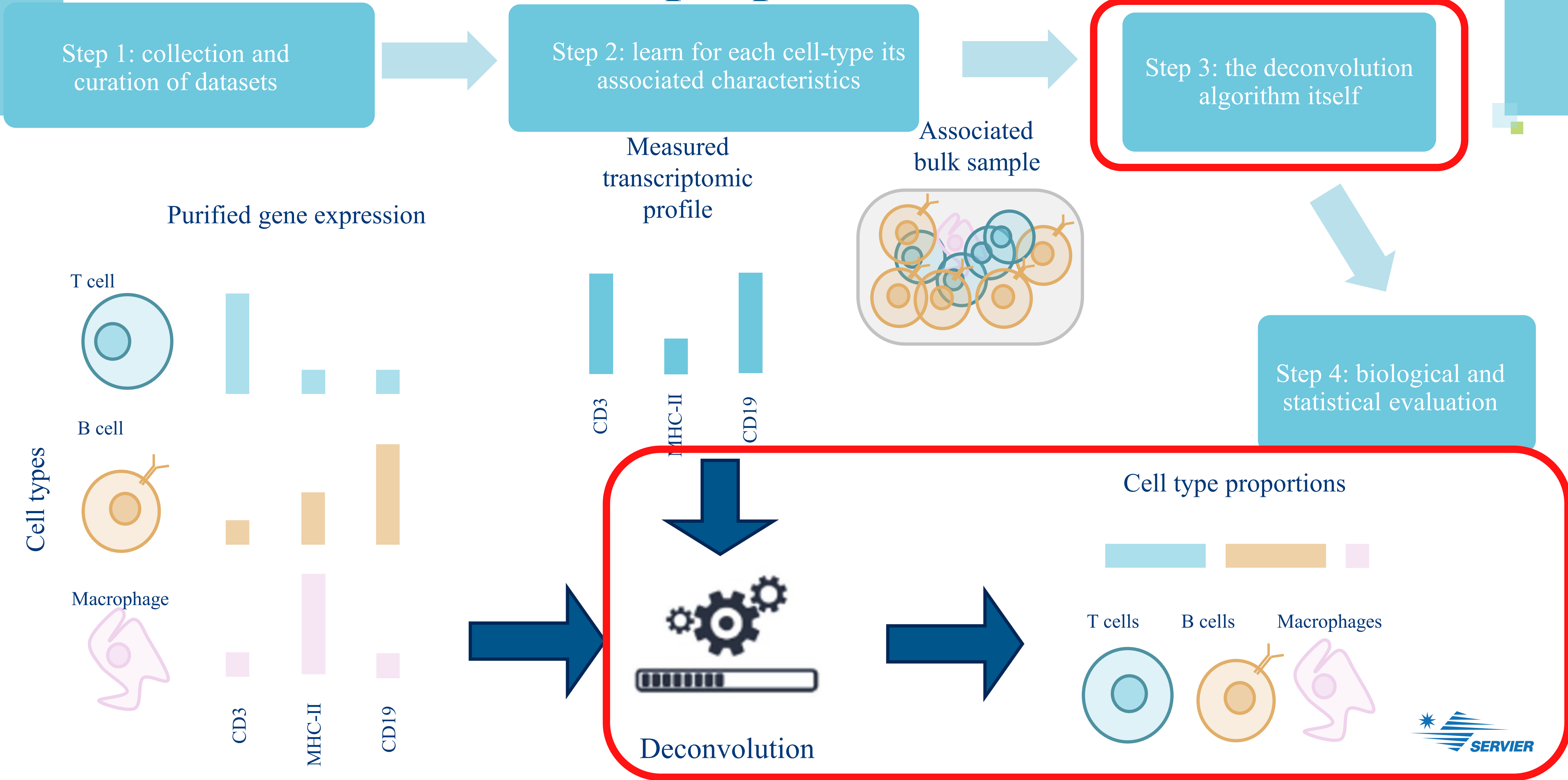
Build a sparse graphical model

$$\forall (i,l) \in V, \quad X_i \perp\!\!\!\perp X_l \Leftrightarrow \rho_{i,l|V\setminus\{i,l\}} = 0$$

$$\rho_{i,l|V\setminus\{i,l\}} = -\frac{\theta_{il}}{\sqrt{\theta_{ii}\theta_{ll}}}$$

If partial correlation is not null between two nodes, we draw an edge between them.

Estimate the cellular proportions



Step 1: collection and curation of datasets

Step 2: learn for each cell-type its associated characteristics

Step 3: the deconvolution algorithm itself

Step 4: biological and statistical evaluation

SERVIER

Step 3: estimate the cellular ratios

$$\begin{pmatrix} x_{1,1} & \cdots & x_{1,J} \\ \vdots & \ddots & \vdots \\ x_{G,1} & \cdots & x_{G,J} \end{pmatrix} \times \underset{\begin{cases} \sum_{j=1}^{J} p_j = 1 \\ \forall j \in \{1, \ldots, J\}, \quad p_j \geq 0 \end{cases}}{} \begin{pmatrix} p_{1,1} & \cdots & p_{1,N} \\ \vdots & \ddots & \vdots \\ p_{J,1} & \cdots & p_{J,N} \end{pmatrix} = \begin{pmatrix} y_{1,1} & \cdots & y_{1,N} \\ \vdots & \ddots & \vdots \\ y_{G,1} & \cdots & y_{G,N} \end{pmatrix}$$

X purified cellular profiles

p cell ratios

Y bulk expression

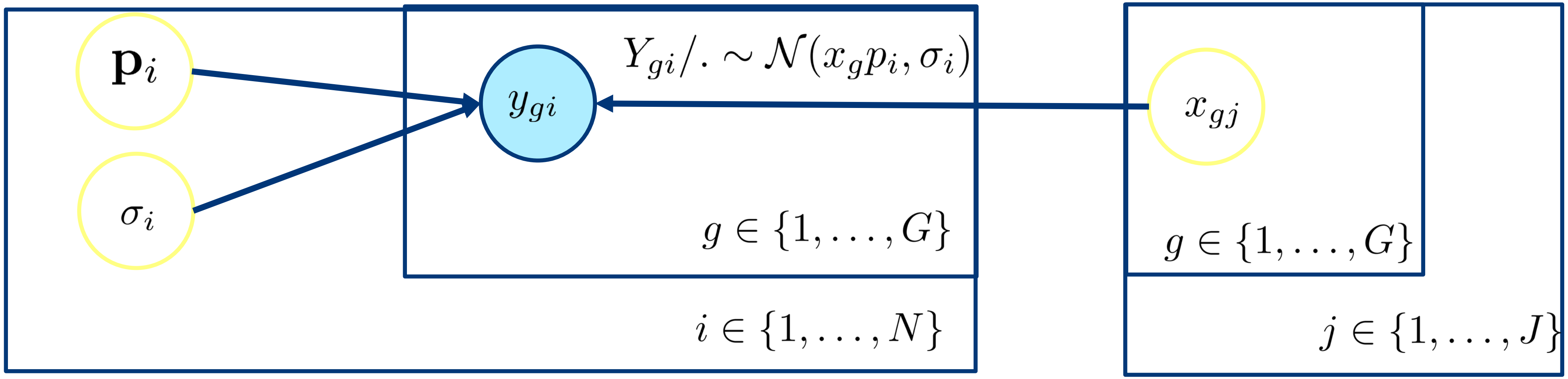
Bulk expression is computed as the weighted linear average of each purified cellular expression profile.

$$\mathbf{y}_i = \mathbf{X} p_i$$

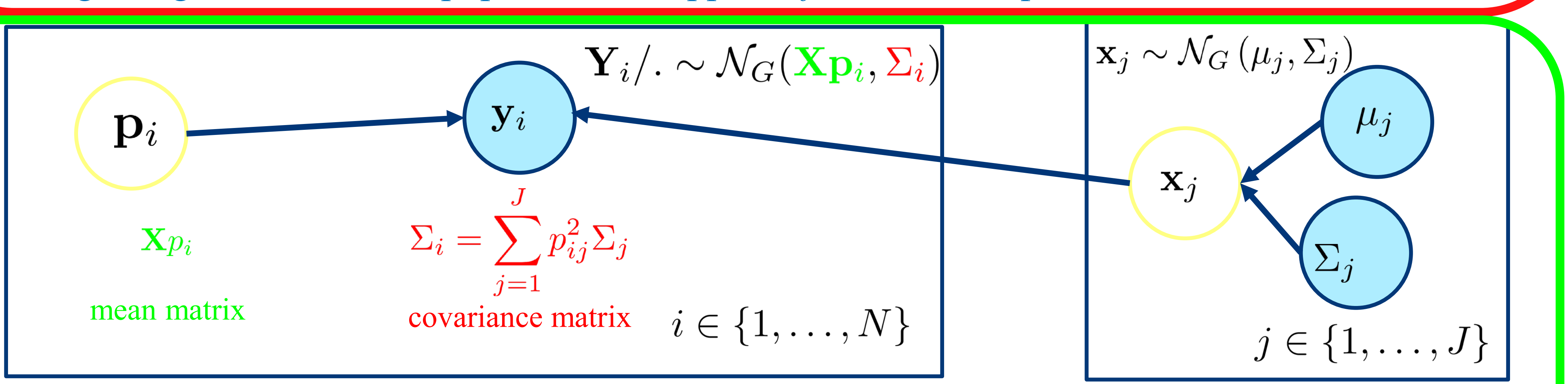
matricial form

$$y_{gi} = \sum_{j=1}^{J} x_{gj} p_{ji}$$

algebraic form



Graphical model of the canonical linear regression modelling. The error between the estimated and the real expression is only accounted by the *uncertainty* σ_i on the measure. The expression of a given gene in each cell population is supposed *fixed* and *independent* from the others.


$\mathbf{Y}_i/. \sim \mathcal{N}_G(\mathbf{X}\mathbf{p}_i, \Sigma_i)$
$\mathbf{p}_i$
$\mathbf{y}_i$
$\mathbf{X}p_i$
mean matrix
$\Sigma_i = \sum_{j=1}^{J} p_{ij}^2 \Sigma_j$
covariance matrix
$i \in \{1, \ldots, N\}$
$\mathbf{x}_j \sim \mathcal{N}_G(\mu_j, \Sigma_j)$
$\mathbf{x}_j$
μ_j
Σ_j
$j \in \{1, \ldots, J\}$

Graphical model of our multivariate modelling: the variability is brought by the individual reference profiles themselves, and the genes *interplay* together.

MLE estimation in the multivariate scenario

$$\hat{p}_i = \arg\min_{\hat{p}_i} ||\mathbf{X}\hat{p}_i - y_i||^2$$

$$\hat{p}_i^{\mathrm{OLS}} = (\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{y}_i$$

With the Gaussian-Markov assumptions, OLS is the best *BLUE* estimator and equal to the MLE estimate.

$$\ell_{\mathbf{y}|\mathbf{X},\Sigma}(\mathbf{p}) = C+\log\left(\det\left(\sum_{j=1}^{J} p_j^2\Sigma_j\right)^{-1}\right)-\frac{1}{2}(\mathbf{y}-\mathbf{X}\mathbf{p})^\top\left(\sum_{j=1}^{J} p_j^2\Sigma_j\right)^{-1}(\mathbf{y}-\mathbf{X}\mathbf{p})$$

Main difficulty in finding the MLE of the log-likelihood function is in inverting the red covariance matrix, making it an intractable analytic problem without further assumption.

$$\begin{cases} p_j & = \frac{e^{p_j}}{\sum_{j=1}^{J-1} e^{p_j}+1}, j < J \\ p_J & = \frac{1}{\sum_{j=1}^{J-1} e^{p_j}+1} \end{cases}$$

- ❑ Descent-gradient based method to learn the MLE.
- ❑ Use of exponentials combine with the sum-to-one ensure that the constraints of non-negativity are enforced during the estimation process

Simulate the distribution: a toy example with two genes and two populations

Generation of random purified cellular expression profile, independently for each individual and each cell population

$$\mathbf{X}_j \sim \mathcal{N}_2(\mu_j, \Sigma_j)$$

$$\begin{array}{ccc} & \text{cell type 1} & \text{cell type 2} \\ \text{gene 1} & 20 & 22 \\ \text{gene 2} & 22 & 20 \end{array}$$

$$\mu_{1:2,1:2}$$

$$\begin{pmatrix} \sigma_1^2 = 1 & \sigma_{12}^n = -0.8 + 0.2n \\ \sigma_{21} = \sigma_{12} & \sigma_2^2 = \{1, 4\} \end{pmatrix}$$

$$\Sigma_{1:2,1:2,1:2} = \left(\Sigma_{1:2,1:2}, \Sigma_{1:2,1:2}\right)^{n \in \{1,\ldots,8\}}$$

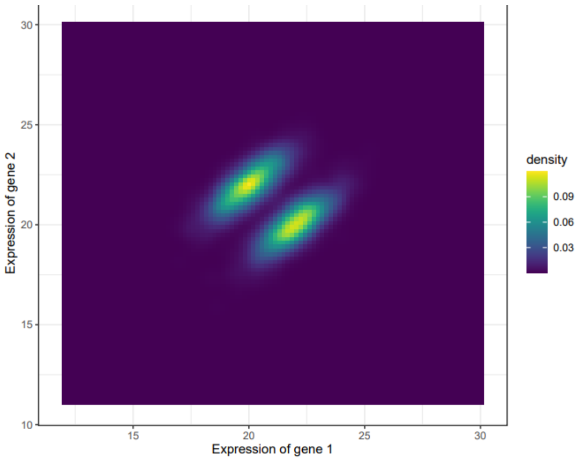
Test several levels of cell proportion disequilibrium:

- ❑ Scenario 1: $p = (0.5, 0.5)$
- ❑ Scenario 2: $p = (0.95, 0.05)$

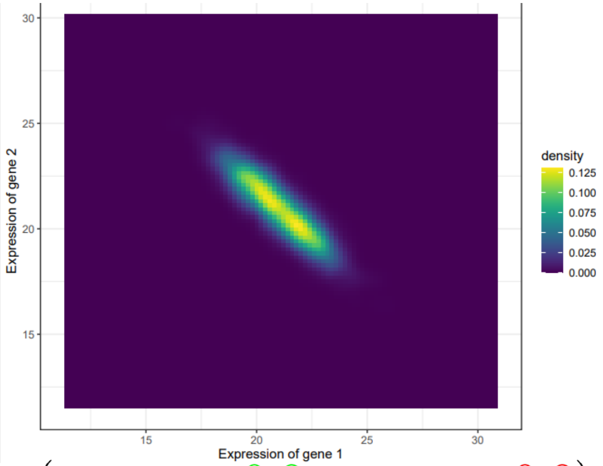
Generation of *N=2000* bulk samples ***Y***

$$\mathbf{y} \sim \mathcal{N}_2(\mathbf{X}p, \Sigma)$$

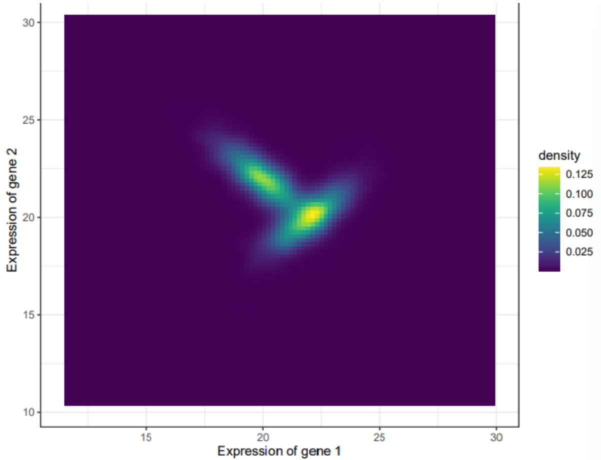
$$\begin{pmatrix} y_{1,1} = \sum_{j=1}^{2} p_j x_{1,j} & \cdots & y_{1,2000} \\ y_{1,2} = \sum_{j=1}^{2} p_j x_{1,j} & \cdots & y_{2,2000} \end{pmatrix}$$



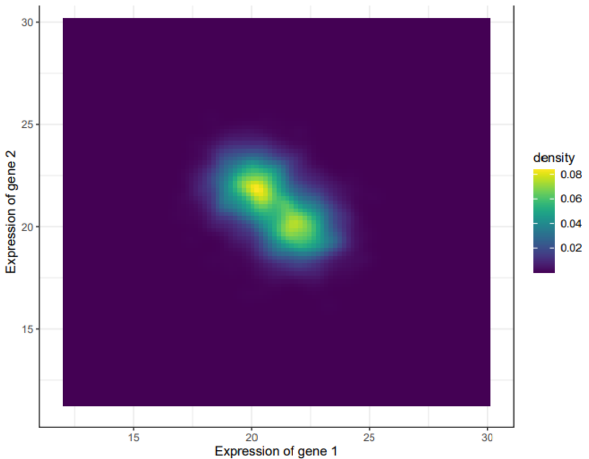
$(\sigma_{121} = 0.8, \sigma_{122} = 0.8)$



$(\sigma_{121} = -0.8, \sigma_{122} = -0.8)$

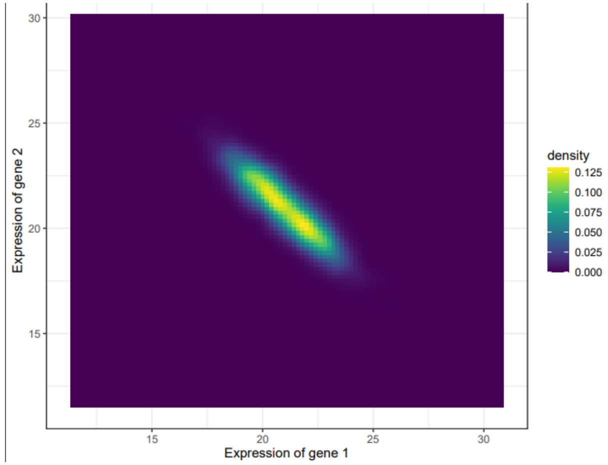


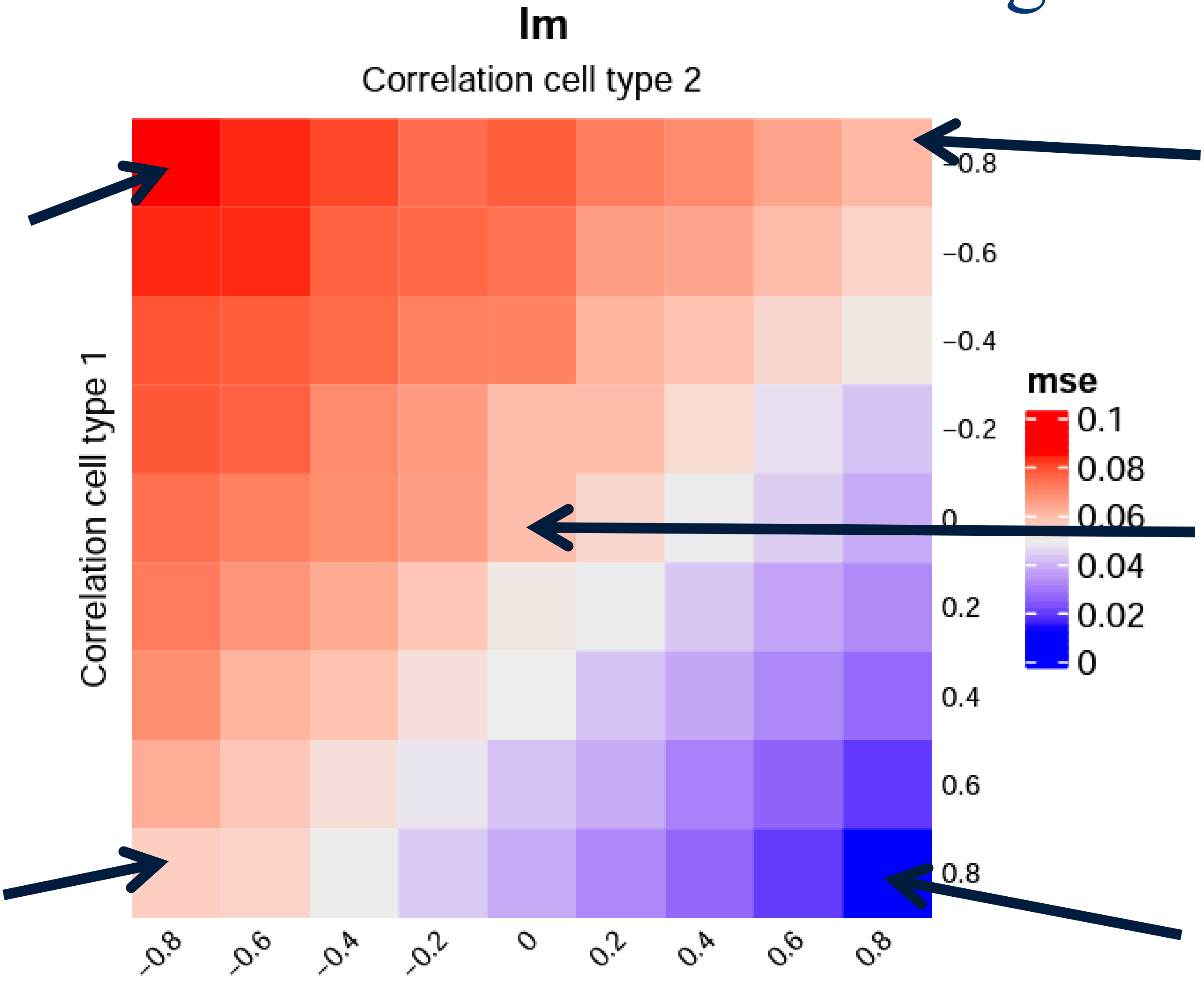
$(\sigma_{121} = -0.8, \sigma_{122} = 0.8)$

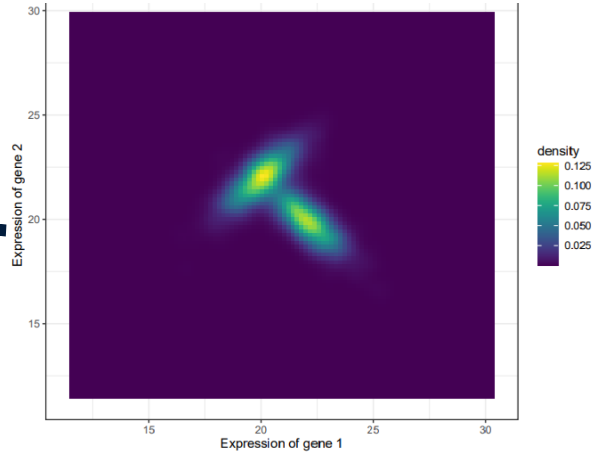


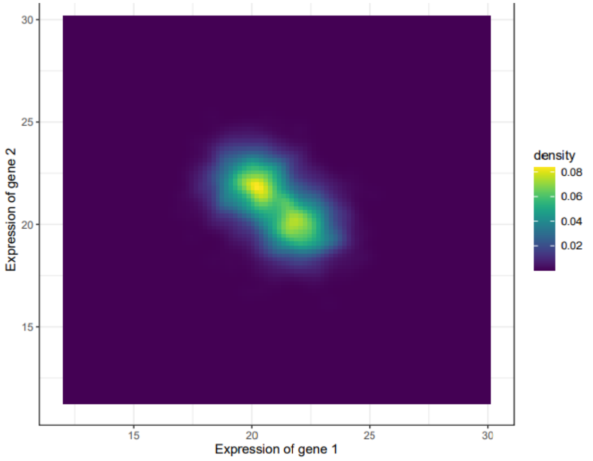
$(\sigma_{121} = 0, \sigma_{122} = 0)$

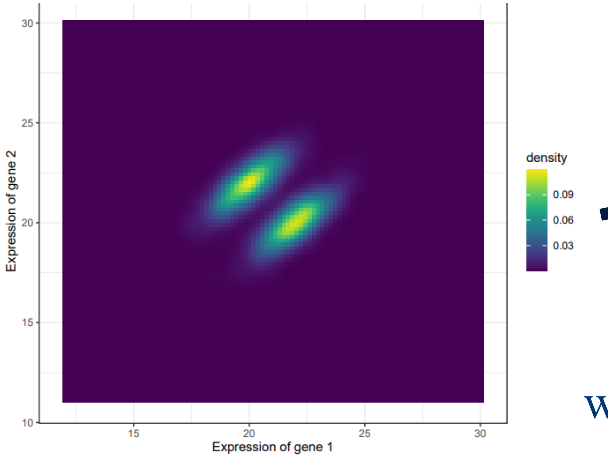
Simulation results with two genes

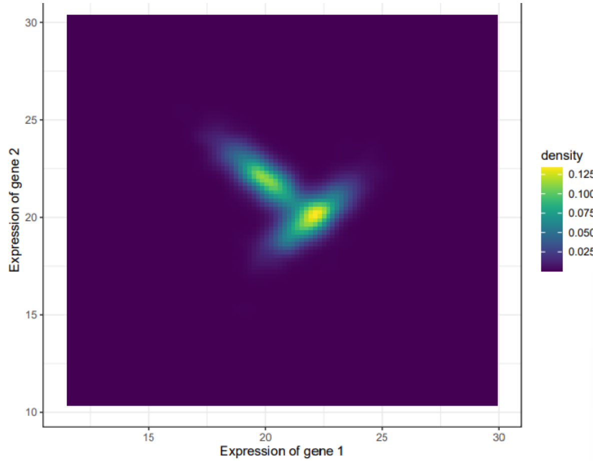






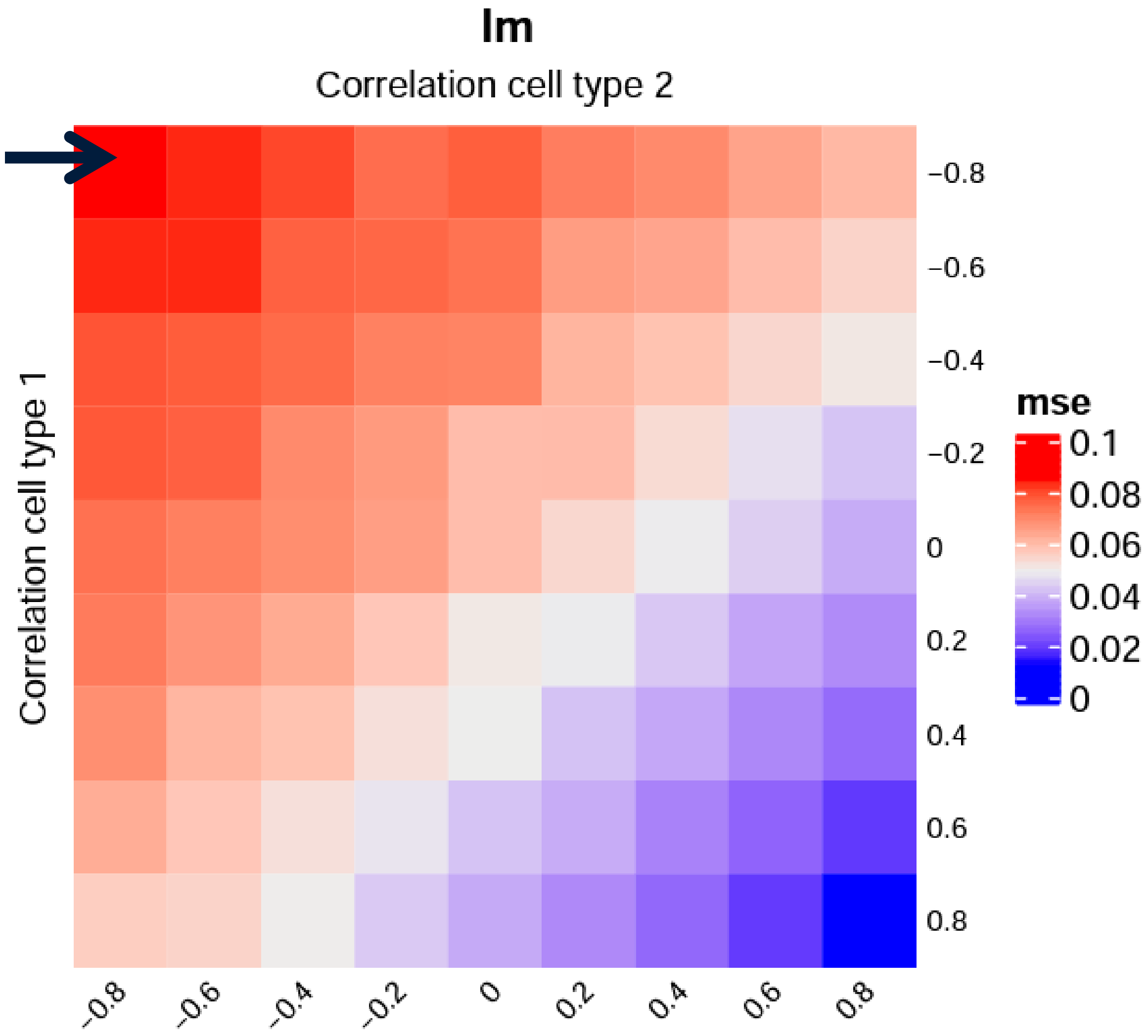




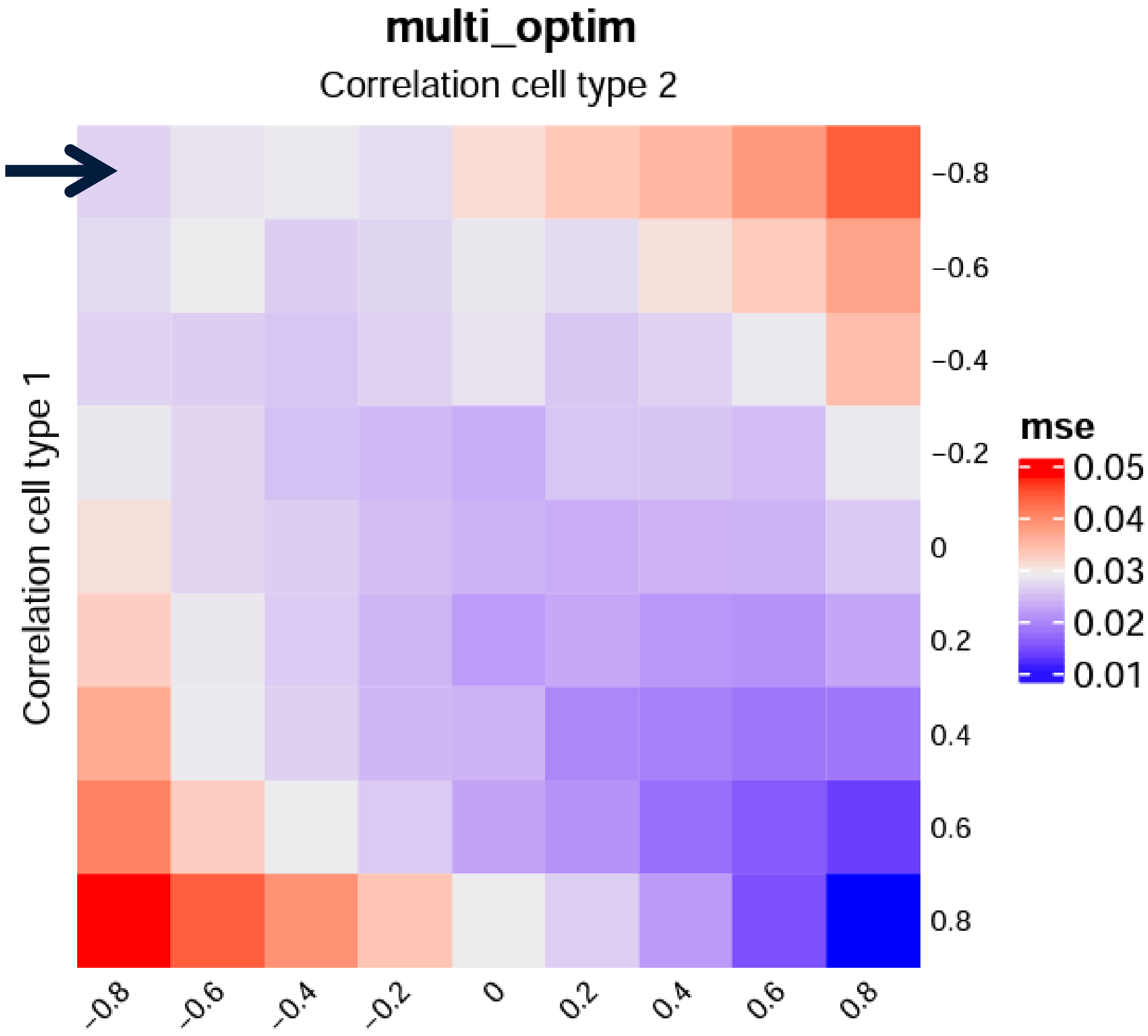


Complex Heatmap (*Gu, 2022*) of the MSE score,
with balanced proportions, high overlapping and homoscedastic genes,
using least-squares method for the estimation of the parameters

Simulation results with two genes

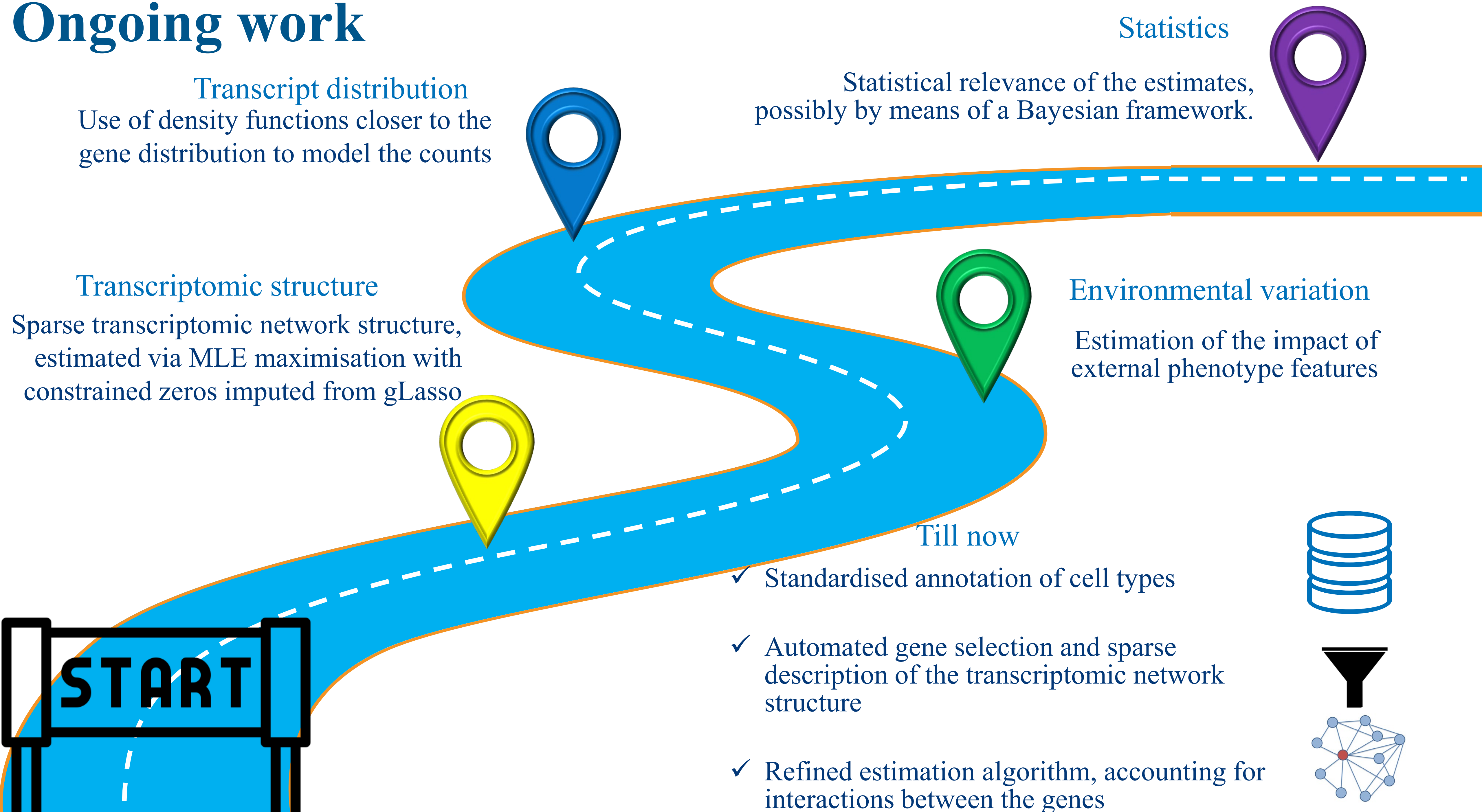


Same Heatmap representation as in the previous slide



Heatmap of the MSE of the estimated ratios, but using this time the covariance information

Ongoing work



Transcript distribution
Use of density functions closer to the gene distribution to model the counts

Statistics

Statistical relevance of the estimates, possibly by means of a Bayesian framework.

Transcriptomic structure

Sparse transcriptomic network structure, estimated via MLE maximisation with constrained zeros imputed from gLasso

Environmental variation

Estimation of the impact of external phenotype features

Till now

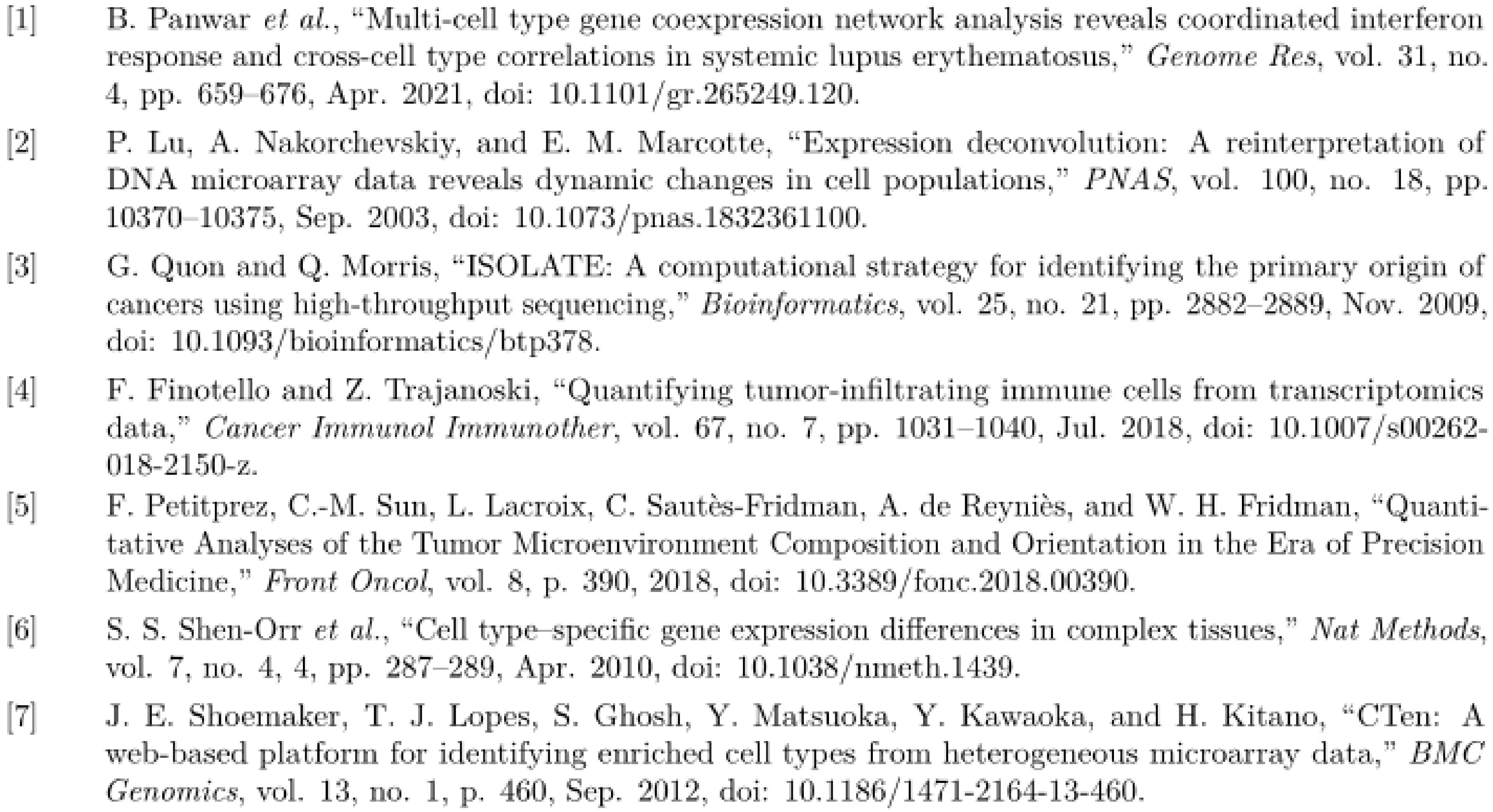
- ✓ Standardised annotation of cell types
- ✓ Automated gene selection and sparse description of the transcriptomic network structure
- ✓ Refined estimation algorithm, accounting for interactions between the genes

Acknowledgement

Thanks for your attention,

A special thought to my tutors from Sorbonne University (LPSM, LIP6) for the theoretical background and to Servier for supplying internal data and automated pipeline for the analysis of transcriptomic data.

References



[1] B. Panwar *et al.*, “Multi-cell type gene coexpression network analysis reveals coordinated interferon response and cross-cell type correlations in systemic lupus erythematosus,” *Genome Res*, vol. 31, no. 4, pp. 659–676, Apr. 2021, doi: 10.1101/gr.265249.120.

[2] P. Lu, A. Nakorchevskiy, and E. M. Marcotte, “Expression deconvolution: A reinterpretation of DNA microarray data reveals dynamic changes in cell populations,” *PNAS*, vol. 100, no. 18, pp. 10370–10375, Sep. 2003, doi: 10.1073/pnas.1832361100.

[3] G. Quon and Q. Morris, “ISOLATE: A computational strategy for identifying the primary origin of cancers using high-throughput sequencing,” *Bioinformatics*, vol. 25, no. 21, pp. 2882–2889, Nov. 2009, doi: 10.1093/bioinformatics/btp378.

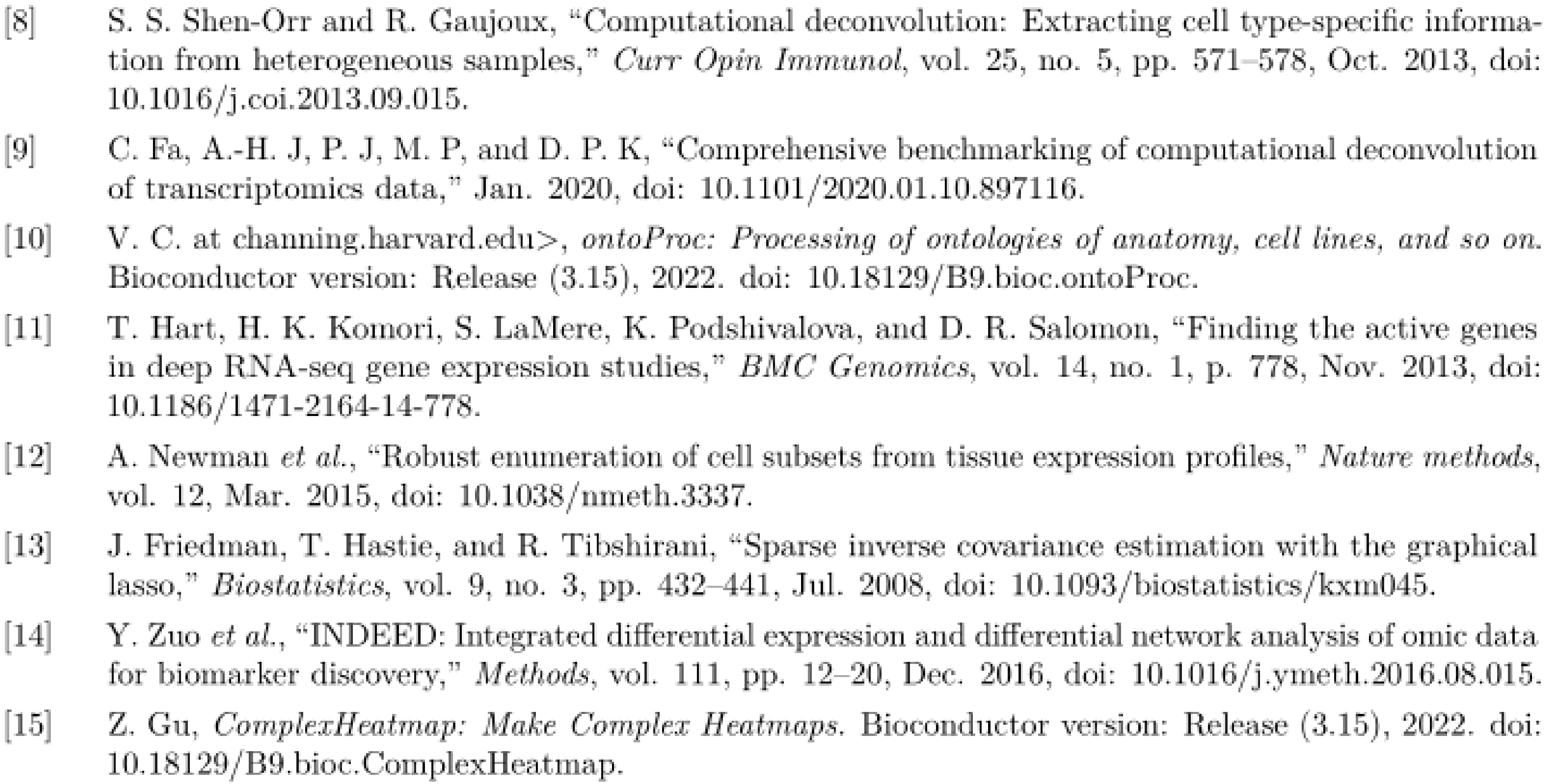
[4] F. Finotello and Z. Trajanoski, “Quantifying tumor-infiltrating immune cells from transcriptomics data,” *Cancer Immunol Immunother*, vol. 67, no. 7, pp. 1031–1040, Jul. 2018, doi: 10.1007/s00262-018-2150-z.

[5] F. Petitprez, C.-M. Sun, L. Lacroix, C. Sautès-Fridman, A. de Reyniès, and W. H. Fridman, “Quantitative Analyses of the Tumor Microenvironment Composition and Orientation in the Era of Precision Medicine,” *Front Oncol*, vol. 8, p. 390, 2018, doi: 10.3389/fonc.2018.00390.

[6] S. S. Shen-Orr *et al.*, “Cell type–specific gene expression differences in complex tissues,” *Nat Methods*, vol. 7, no. 4, 4, pp. 287–289, Apr. 2010, doi: 10.1038/nmeth.1439.

[7] J. E. Shoemaker, T. J. Lopes, S. Ghosh, Y. Matsuoka, Y. Kawaoka, and H. Kitano, “CTen: A web-based platform for identifying enriched cell types from heterogeneous microarray data,” *BMC Genomics*, vol. 13, no. 1, p. 460, Sep. 2012, doi: 10.1186/1471-2164-13-460.

References



[8] S. S. Shen-Orr and R. Gaujoux, "Computational deconvolution: Extracting cell type-specific information from heterogeneous samples," *Curr Opin Immunol*, vol. 25, no. 5, pp. 571–578, Oct. 2013, doi: 10.1016/j.coi.2013.09.015.

[9] C. Fa, A.-H. J, P. J, M. P, and D. P. K, "Comprehensive benchmarking of computational deconvolution of transcriptomics data," Jan. 2020, doi: 10.1101/2020.01.10.897116.

[10] V. C. at channing.harvard.edu>, *ontoProc: Processing of ontologies of anatomy, cell lines, and so on.* Bioconductor version: Release (3.15), 2022. doi: 10.18129/B9.bioc.ontoProc.

[11] T. Hart, H. K. Komori, S. LaMere, K. Podshivalova, and D. R. Salomon, "Finding the active genes in deep RNA-seq gene expression studies," *BMC Genomics*, vol. 14, no. 1, p. 778, Nov. 2013, doi: 10.1186/1471-2164-14-778.

[12] A. Newman *et al.*, "Robust enumeration of cell subsets from tissue expression profiles," *Nature methods*, vol. 12, Mar. 2015, doi: 10.1038/nmeth.3337.

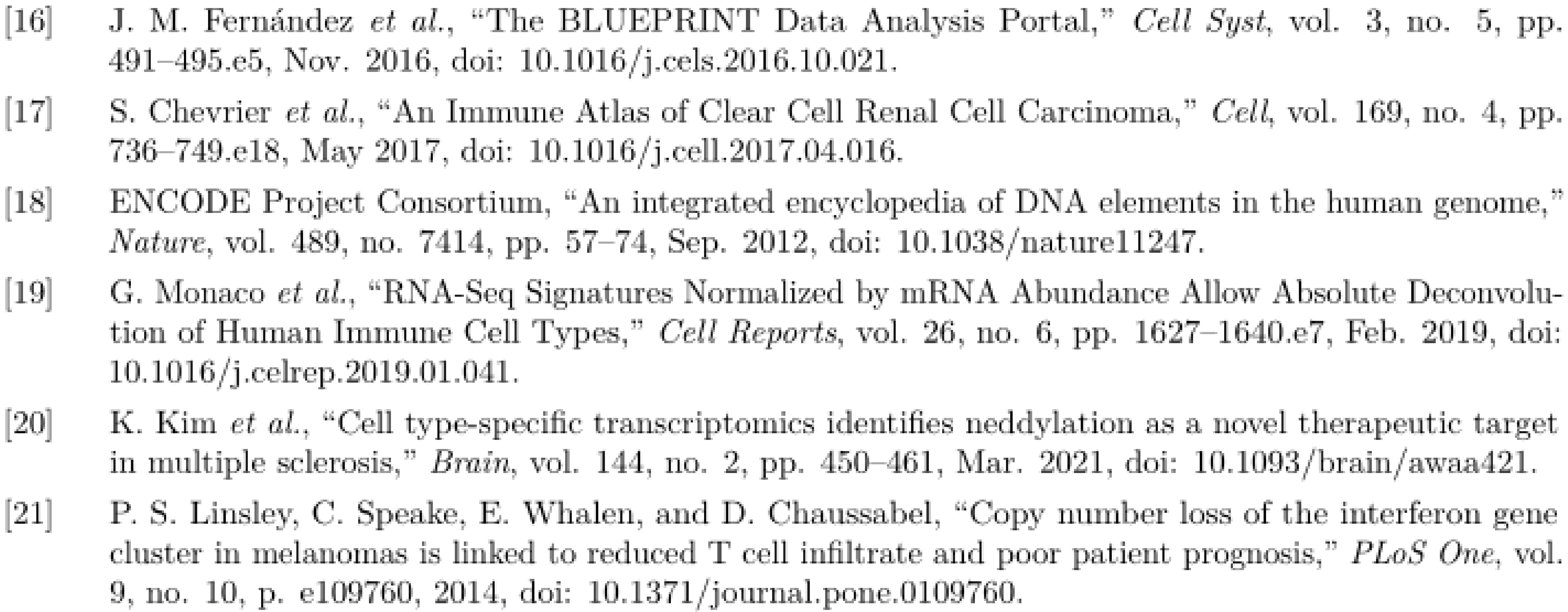
[13] J. Friedman, T. Hastie, and R. Tibshirani, "Sparse inverse covariance estimation with the graphical lasso," *Biostatistics*, vol. 9, no. 3, pp. 432–441, Jul. 2008, doi: 10.1093/biostatistics/kxm045.

[14] Y. Zuo *et al.*, "INDEED: Integrated differential expression and differential network analysis of omic data for biomarker discovery," *Methods*, vol. 111, pp. 12–20, Dec. 2016, doi: 10.1016/j.ymeth.2016.08.015.

[15] Z. Gu, *ComplexHeatmap: Make Complex Heatmaps.* Bioconductor version: Release (3.15), 2022. doi: 10.18129/B9.bioc.ComplexHeatmap.

Theory and packages

References



[16] J. M. Fernández *et al.*, "The BLUEPRINT Data Analysis Portal," *Cell Syst*, vol. 3, no. 5, pp. 491–495.e5, Nov. 2016, doi: 10.1016/j.cels.2016.10.021.

[17] S. Chevrier *et al.*, "An Immune Atlas of Clear Cell Renal Cell Carcinoma," *Cell*, vol. 169, no. 4, pp. 736–749.e18, May 2017, doi: 10.1016/j.cell.2017.04.016.

[18] ENCODE Project Consortium, "An integrated encyclopedia of DNA elements in the human genome," *Nature*, vol. 489, no. 7414, pp. 57–74, Sep. 2012, doi: 10.1038/nature11247.

[19] G. Monaco *et al.*, "RNA-Seq Signatures Normalized by mRNA Abundance Allow Absolute Deconvolution of Human Immune Cell Types," *Cell Reports*, vol. 26, no. 6, pp. 1627–1640.e7, Feb. 2019, doi: 10.1016/j.celrep.2019.01.041.

[20] K. Kim *et al.*, "Cell type-specific transcriptomics identifies neddylation as a novel therapeutic target in multiple sclerosis," *Brain*, vol. 144, no. 2, pp. 450–461, Mar. 2021, doi: 10.1093/brain/awaa421.

[21] P. S. Linsley, C. Speake, E. Whalen, and D. Chaussabel, "Copy number loss of the interferon gene cluster in melanomas is linked to reduced T cell infiltrate and poor patient prognosis," *PLoS One*, vol. 9, no. 10, p. e109760, 2014, doi: 10.1371/journal.pone.0109760.

Outline

01 Analysing the biological medium

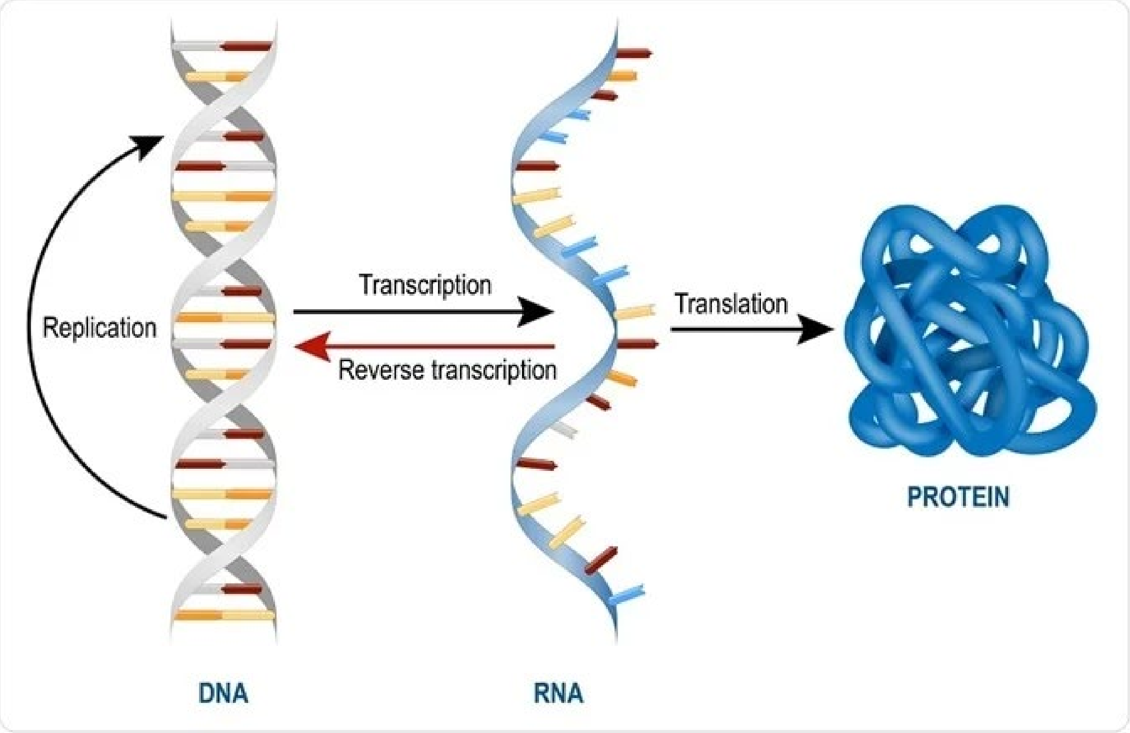
02 General principle of deconvolution algorithms

03 Standard deconvolution pipeline

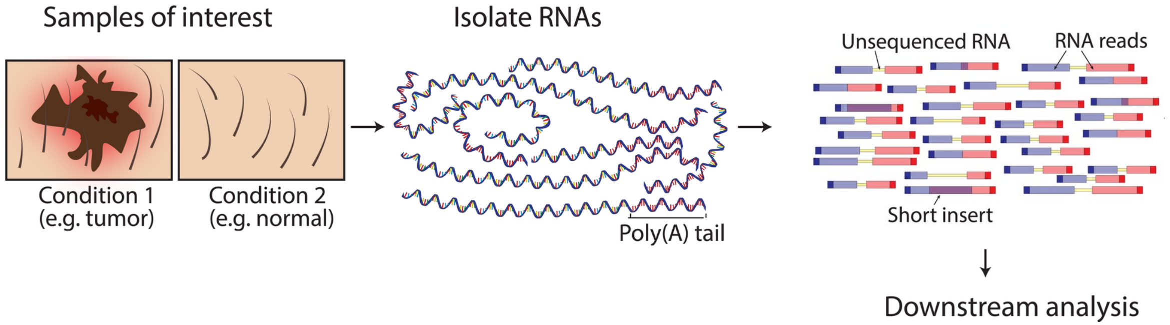
04 Multivariate extension to standard deconvolution algorithms

05 Numerical simulation and future development

What is transcriptomics?



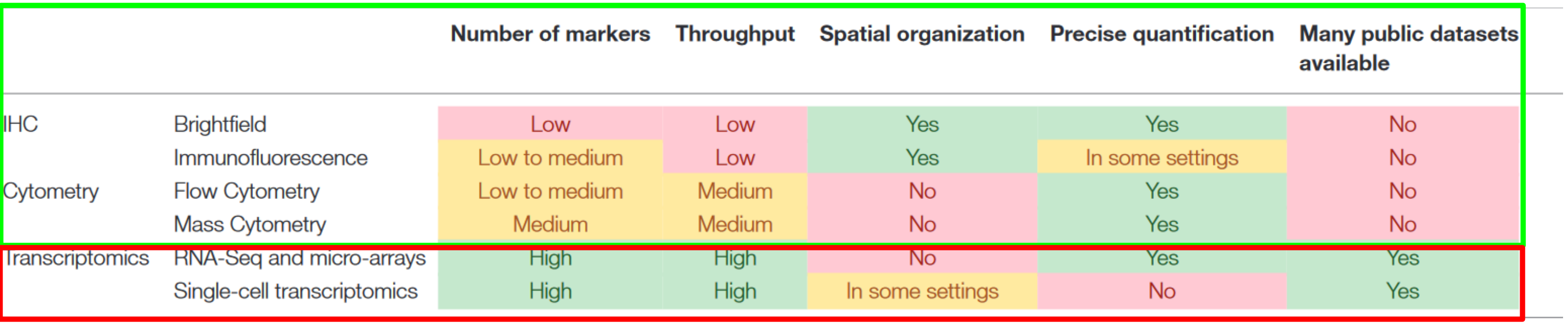
Quantifying mRNA



From transcriptomics to biological insights

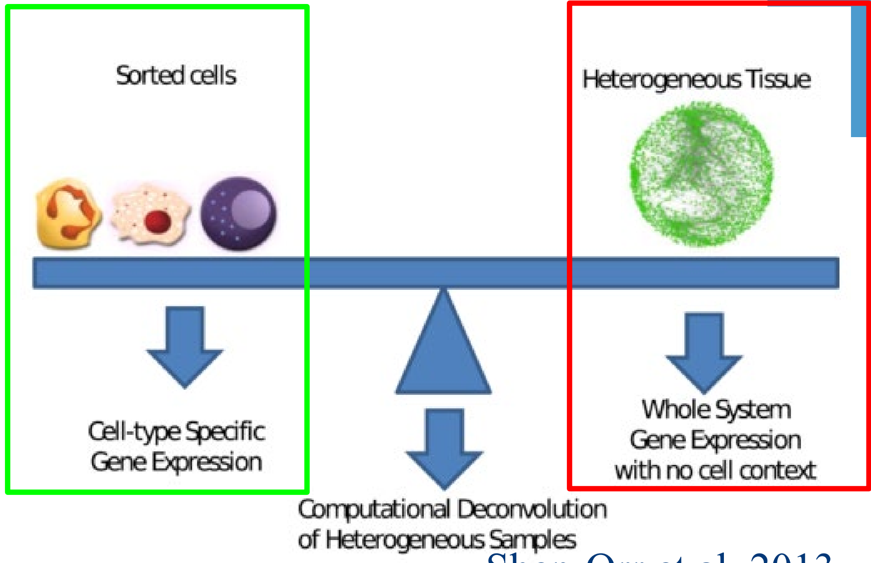
Shen-Orr et al, NIH, 2013

Physical methods to analyse the biological medium



		Number of markers	Throughput	Spatial organization	Precise quantification	Many public datasets available
IHC	Brightfield	Low	Low	Yes	Yes	No
	Immunofluorescence	Low to medium	Low	Yes	In some settings	No
Cytometry	Flow Cytometry	Low to medium	Medium	No	Yes	No
	Mass Cytometry	Medium	Medium	No	Yes	No
Transcriptomics	RNA-Seq and micro-arrays	High	High	No	Yes	Yes
	Single-cell transcriptomics	High	High	In some settings	No	Yes

Petitprez et al., 2018


Sorted cells
Heterogeneous Tissue
Cell-type Specific Gene Expression
Whole System Gene Expression with no cell context
Computational Deconvolution of Heterogeneous Samples

Shen-Orr et al, 2013

- IHC methods can capture spatial organization but have a low throughput, and can’t discriminate strongly correlated celltypes
- FACS enable precise quantification but have a small throughput and are intrusive
- RNA-Seq and micro-array can analyze expression of many markers, but do not capture the complex sources of their variation
- Before numerical deconvolution, dilemma between either characterising the individual cell populations or getting a whole transcriptomic overview.

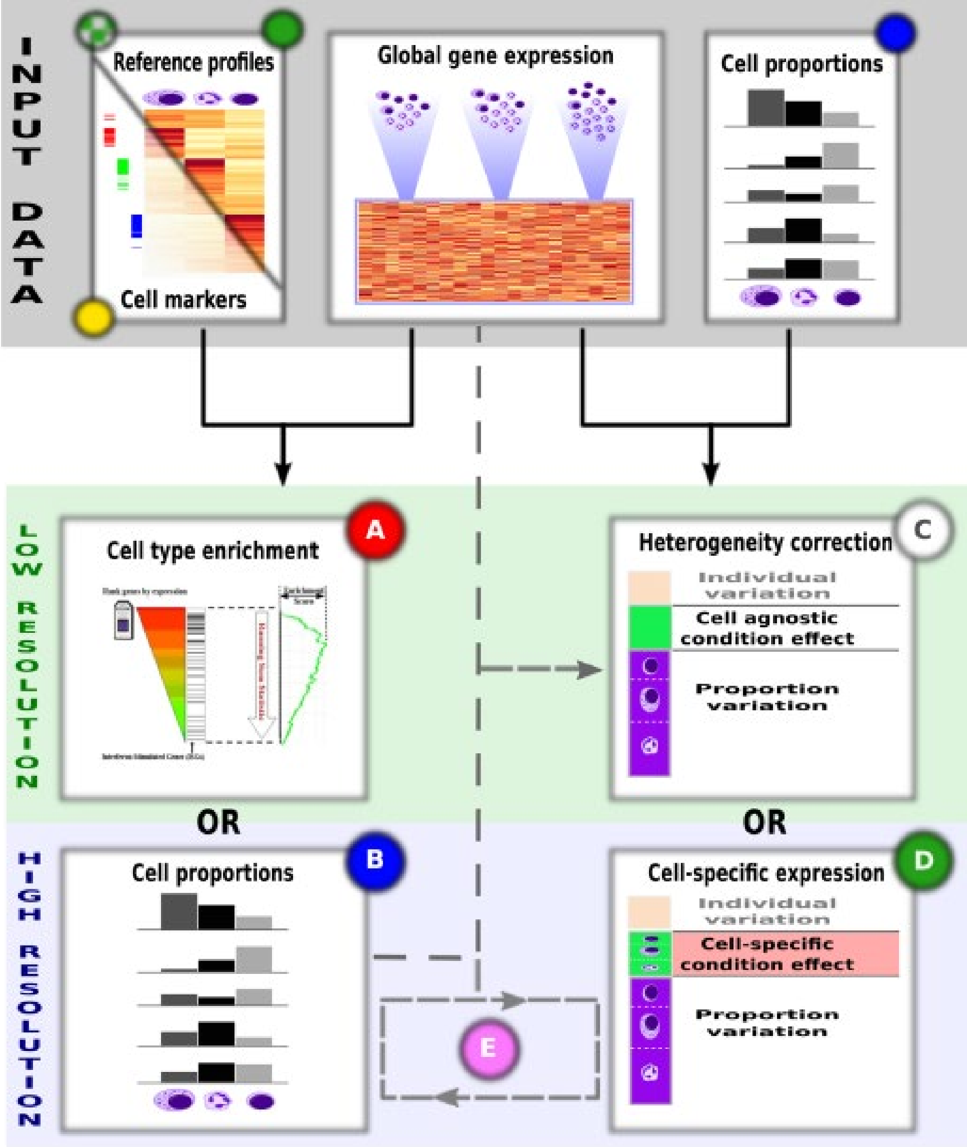
Deconvolution classes

Partial deconvolution

- Estimate the ratios *p* for all individuals with the purified cell signature **X** and bulk mixture **y.**
- Try to infer cell specific expression profiles **X** based on **p** and **y.**

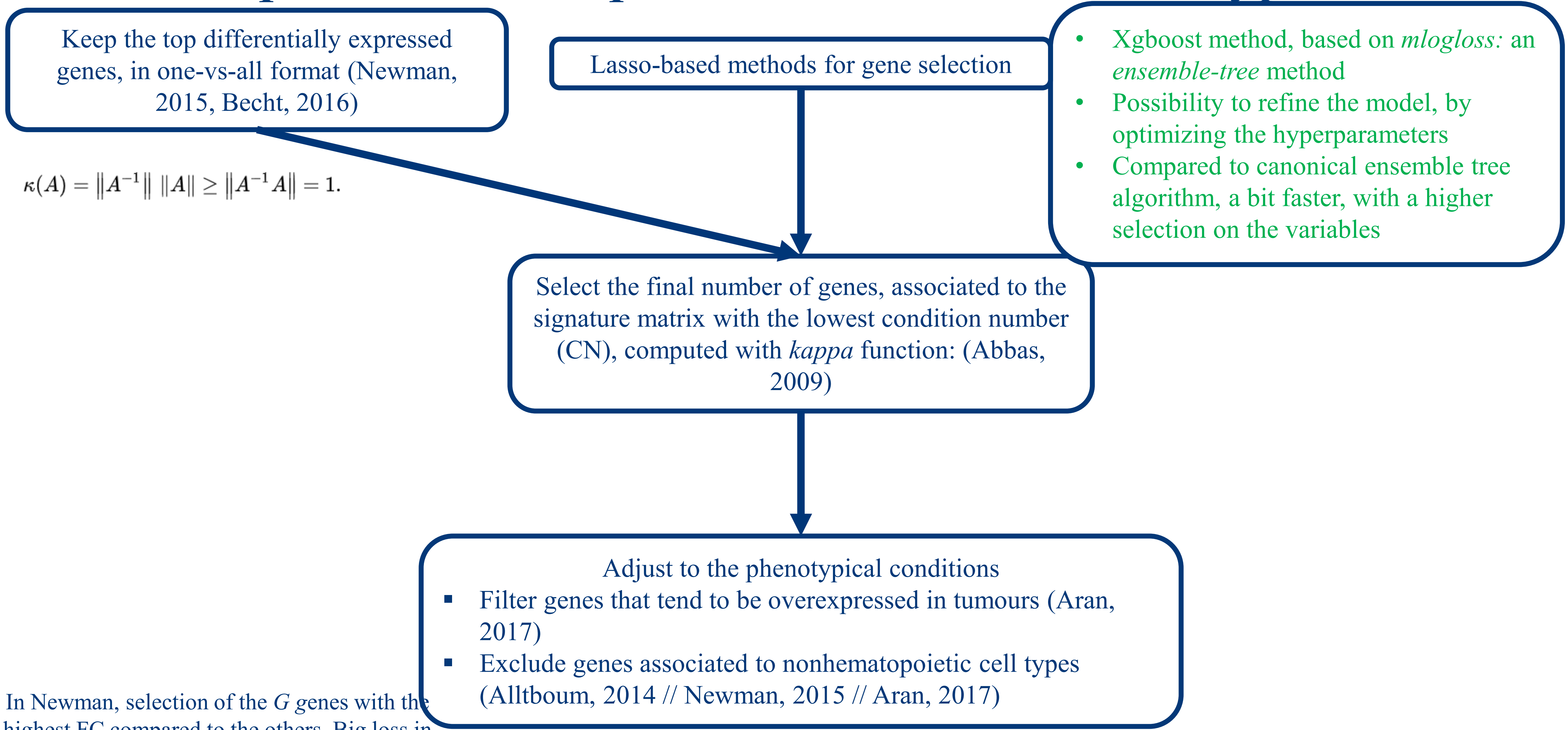
Complete deconvolution

- Try to infer alternatively both **p** and **X** (unsupervised, reference-free methods). Undetermined problem without prior.



Shen-Orr et al, 2013

Step 2: learn the sparse GGM for each cell type



Keep the top differentially expressed genes, in one-vs-all format (Newman, 2015, Becht, 2016)

Lasso-based methods for gene selection

- Xgboost method, based on *mlogloss: an ensemble-tree* method
- Possibility to refine the model, by optimizing the hyperparameters
- Compared to canonical ensemble tree algorithm, a bit faster, with a higher selection on the variables

$$\kappa(A) = \left\|A^{-1}\right\| \, \|A\| \geq \left\|A^{-1}A\right\| = 1.$$

Select the final number of genes, associated to the signature matrix with the lowest condition number (CN), computed with *kappa* function: (Abbas, 2009)

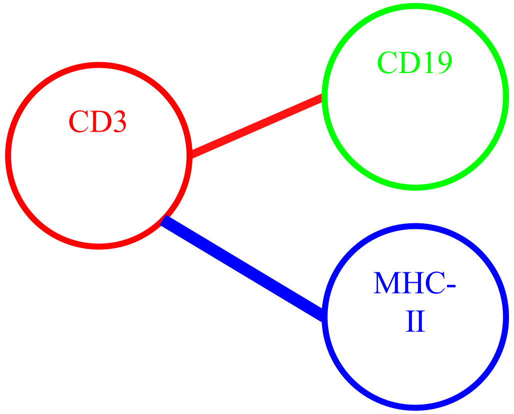
Adjust to the phenotypical conditions

- Filter genes that tend to be overexpressed in tumours (Aran, 2017)
- Exclude genes associated to nonhematopoietic cell types (Alltboum, 2014 // Newman, 2015 // Aran, 2017)

In Newman, selection of the *G* genes with the highest FC compared to the others. Big loss in kappa is likely to correspond to
the inclusion of a gene setting apart a population

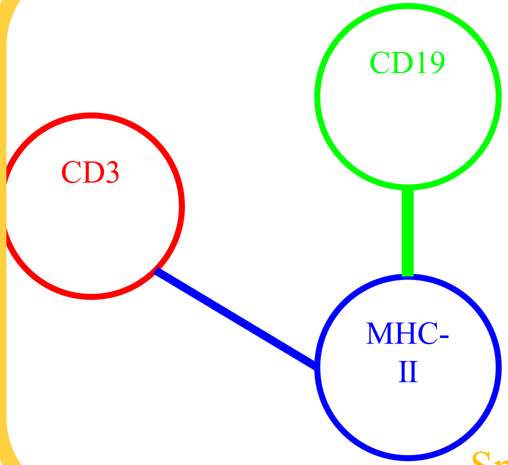
Sparse feature selection

Step 2: learn the sparse GGM for each cell type



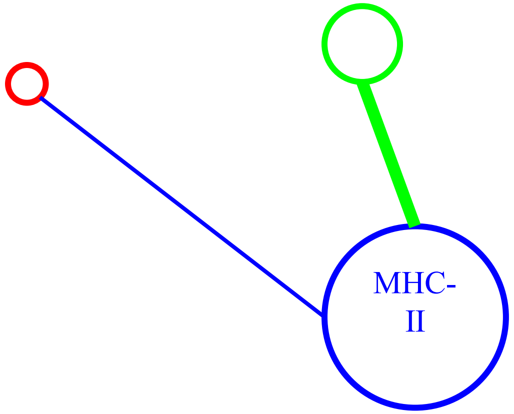
$$\begin{array}{cccc} \text{CD3} & \text{CD19} & \text{MHC-II} & \\ \sigma_1 & 0.5 & 0.8 & \text{CD3} \\ & \sigma_2 & 0 & \text{CD19} \\ & & \sigma_3 & \text{MHC-II} \end{array}$$

Sparse network in cell type A



$$\begin{array}{cccc} \text{CD3} & \text{CD19} & \text{MHC-II} & \\ \sigma_1 & 0 & 0.5 & \text{CD3} \\ & \sigma_2 & 0.8 & \text{CD19} \\ & & \sigma_3 & \text{MHC-II} \end{array}$$

Sparse network in cell type B



$$\begin{array}{cccc} \text{CD3} & \text{CD19} & \text{MHC-II} & \\ \sigma_1 & 0 & 0.3 & \text{CD3} \\ & \sigma_2 & -0.8 & \text{CD19} \\ & & \sigma_3 & \text{MHC-II} \end{array}$$

$$\Omega = \{\omega_{gl},\ \text{where } \Delta_{gl} = \rho_{gl}^A - \rho g l^B \neq 0\}$$

Differential network of both conditions

- Size of each gene is proportional to its activity score, given by summing the *z*-scores of its neighbourhood
- *W*-Indeed is a weighted extension, accounting for distinct datasets

Zoom on the INDEED (*Zuo et al, 2016*) algorithm

Step 3: estimate the cellular ratios

$$\begin{pmatrix} x_{1,1} & \cdots & x_{1,J} \\ \vdots & \ddots & \vdots \\ x_{G,1} & \cdots & x_{G,J} \end{pmatrix} \times \begin{pmatrix} p_{1,1} & \cdots & p_{1,N} \\ \vdots & \ddots & \vdots \\ p_{J,1} & \cdots & p_{J,N} \end{pmatrix} = \begin{pmatrix} y_{1,1} & \cdots & y_{1,N} \\ \vdots & \ddots & \vdots \\ y_{G,1} & \cdots & y_{G,N} \end{pmatrix}$$

$$\begin{cases} \sum_{j=1}^{J} p_j = 1 \\ \forall j \in \{1, \dots, J\}, \quad p_j \geq 0 \end{cases}$$

X purified cellular profiles

p cell ratios

Y bulk expression

$$\hat{p_i} = \arg\min_{\hat{p_i}} ||\mathbf{X}\hat{p_i} - y_i||^2 = \sum_{g=1}^{G} \left(y_{gi} - \sum_{j=1}^{J} x_{gj}\hat{p_{ji}} \right)$$

Objective: minimize the squared distance of the *residuals* (difference between the physical and estimated gene expression)

$$\hat{p}_i^{\text{OLS}} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}_i$$

With the Gaussian-Markov assumptions, OLS is the best *BLUE* estimator (+ uniqueness of the minimal estimate), and confounded with the MLE estimate

$$\mathbf{y}_i = \mathbf{X} p_i + \epsilon_i$$

Bulk expression is computed as the weighted linear average of each purified cellular expression profile

$$\epsilon_i \sim \mathcal{N}(0, \sigma)$$

Variability is only brought by the uncertainty on the measure, assuming to follow a white Gaussian noise.

Step 3: choice of the deconvolution algorithm

Marker-based

- Abundance scores (dtangle algorithm, …)
- Enrichment scores (new methods for computing it …)

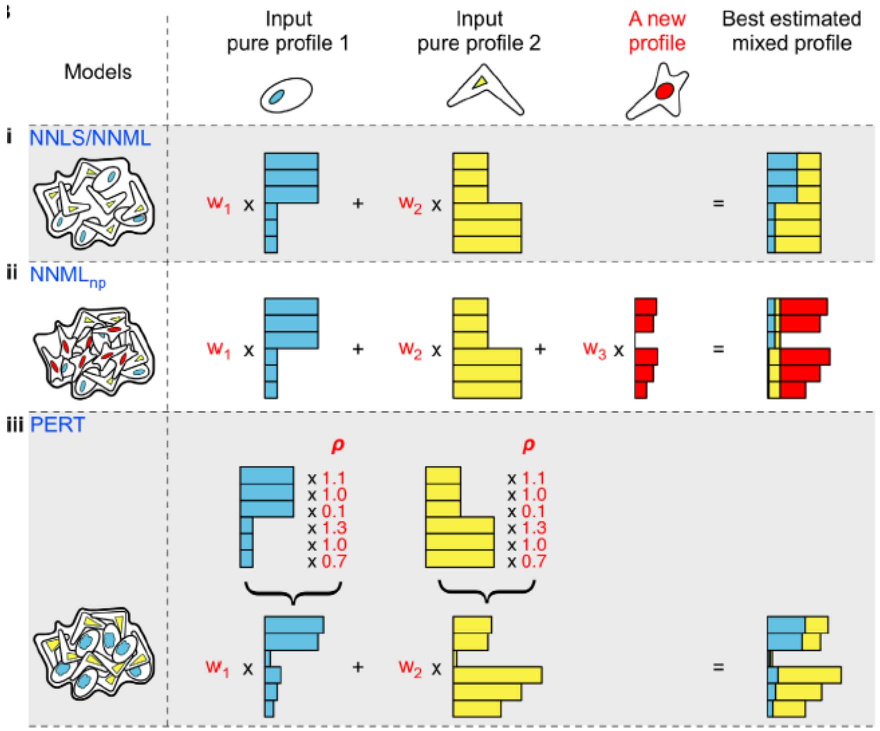
regression

- Robust regression to deal with outliers
- Regression methods relaxing the Gaussian Markow assumptions
- Model counts with NBs or Poisson regression

probabilistic

- Variational EM algorithm with latent environmental variation and prior information on the ratio
- Account for bias transcription

Estimate the ratios from the reference signature and bulk mixture



Qiao et al, 2012

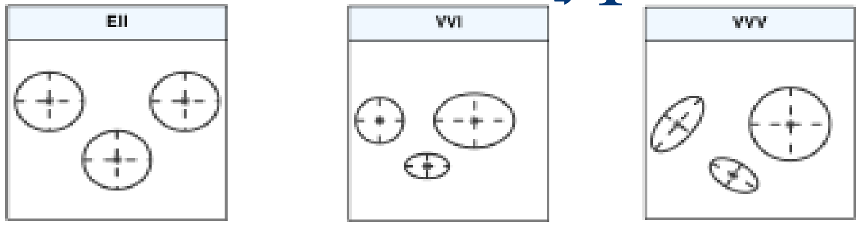
Step 2: learn the sparse GGM for each cell type

Multivariate gaussian distribution

$$\mathbf{X}_{1:G,j} \sim \mathcal{N}_G(\boxed{\mu_j}, \boxed{\mathbf{\Sigma}_j})$$

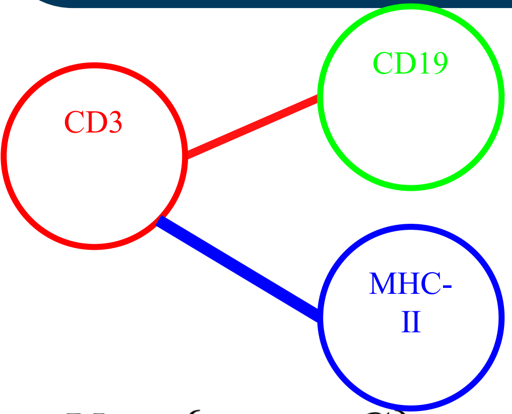
$\mu = \mathrm{E}(\mathbf{X})$
Mean vector

$\mathbf{\Sigma}_{i,l} = \mathrm{Cov}(X_i, X_l), \forall 1 \leq i, l \leq G$
Covariance matrix



Spherical *Diagonal* *Ellipsoidal*

$$\Sigma_j = \lambda_j D_j A_j D_j^\top$$



$$\begin{array}{cc} \begin{array}{ccc} \text{CD3} & \text{CD19} & \text{MHC-II} \end{array} & \\ \begin{pmatrix} \sigma_1 & 0.5 & 0.8 \\ & \sigma_2 & 0 \\ & & \sigma_3 \end{pmatrix} & \begin{array}{c} \text{CD3} \\ \text{CD19} \\ \text{MHC-II} \end{array} \end{array}$$

$V = \{1, \dots, G\}$

$E = \{i, l \in V^2, i \neq l\}$

Estimate a sparse covariance structure using gLasso (*Friedman et al, 2008)* algorithm

Precision matrix: the inverse of the covariance matrix

$$\Theta = (\theta_{il}, \quad (i,l) \in \{1, \dots, G\}) = \Sigma^{-1}$$

Its corresponding sparse estimate:

$$\Theta_{\text{Lasso}} = \arg\max_{\Theta} \left(\log(\det(\Theta)) - \mathrm{Tr}(S\Theta) - \lambda||\Theta||_1\right)$$

With $\lambda = 0$, returns the MLE estimate of the precision matrix.

Nb: possibility to set a prior weight during the gLasso estimation on the connections between the genes (PPI is often use for that purpose)

Build a sparse graphical model

$$\forall (i,l) \in V, \quad X_i \perp\!\!\!\perp X_l \Leftrightarrow \rho_{i,l|V\setminus\{i,l\}} = 0$$

$$\rho_{i,l|V\setminus\{i,l\}} = -\frac{\theta_{il}}{\sqrt{\theta_{ii}\theta_{ll}}}$$

if partial correlation is not null between two nodes, an edge connecting them is drawn.

Framework of the multivariate probalistic model

- We can show that the *conditional distribution* of the bulk mixture follows itself a *multivariate Gaussian distribution*, with that modelling framework.

- We combine assumption of *independence* between the cell types with the *invariant* property of Gaussian distributions under *affine* transformation.

$$\mathbf{y}_i/\mathbf{X} \sim \mathcal{N}_G(\mu_i, \Sigma_i)$$

$$\mu_i = \mathbf{X}p_i \qquad \Sigma_i = \sum_{j=1}^{J} p_{ij}^2 \Sigma_j$$

mean matrix

covariance matrix

Step 1: ***X*** is drawn independantly from a multivariate Gaussian distribution for each cell type

$$\mathbf{X}_j \sim \mathcal{N}_G\left(\hat{\mu_j}, \hat{\mathbf{\Sigma}}_j\right)$$

$\hat{\mu}_j$ is the average gene expression in cell type *j* (usual input of partial deconvolution algorithms)

$\hat{\mathbf{\Sigma}}_j$ is the *plugged-in* sparse covariance matrix, estimated via *glasso* or constrained MLE estimation

Step 2: Reconstitute *Y*, the bulk mixture, by summing the weighted contribution of each cellular expression profile.

$$\mathbf{y}_i = \mathbf{X}p_i$$

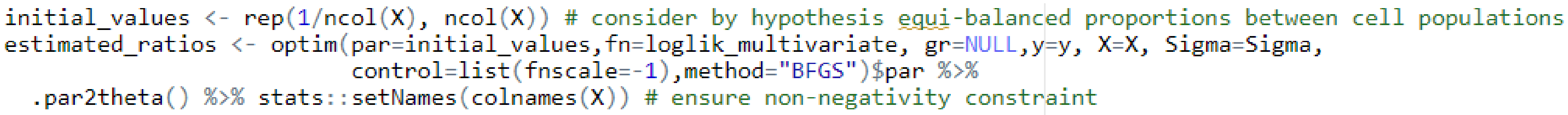
matrix form

$$y_{gi} = \sum_{j=1}^{J} x_{gj}p_{ji}$$

algebraic form

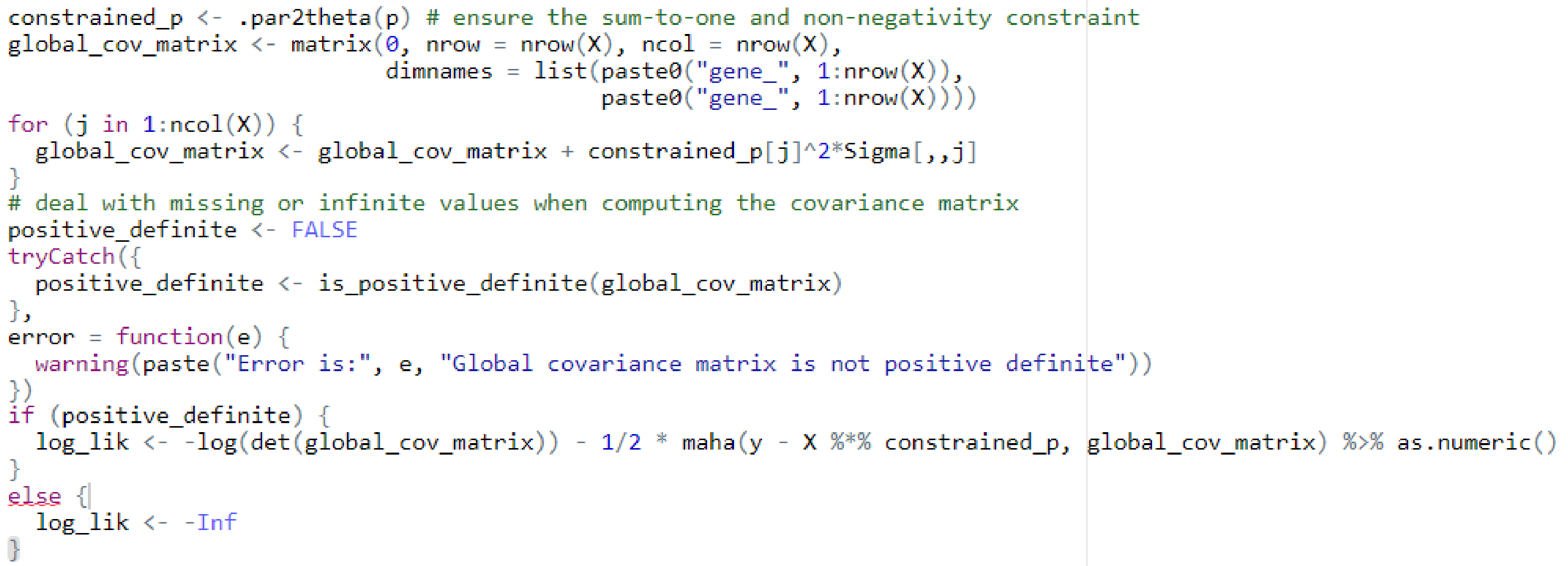
Practical imputation of the MLE estimation using optim function and gradient descent

General optimisation function, using BFGS method
(fnscale is set to -1, as it's a problem of maximisation)



```
initial_values <- rep(1/ncol(X), ncol(X)) # consider by hypothesis equi-balanced proportions between cell populations
estimated_ratios <- optim(par=initial_values,fn=loglik_multivariate, gr=NULL,y=y, X=X, Sigma=Sigma,
                          control=list(fnscale=-1),method="BFGS")$par %>%
  .par2theta() %>% stats::setNames(colnames(X)) # ensure non-negativity constraint
```

- The log-likelihood of the conditional distribution of the observed samples (which reveals to follow a multivariate Gaussian distribution) is given by function *loglik_multivariate*.
- We reparametrize the learnt estimates, *p*, at each iteration step, to enforce the positivity and sum-to-one constraints.
- With two components, we should add *optimize* function, as better fitted for univariate estimation of parameter.



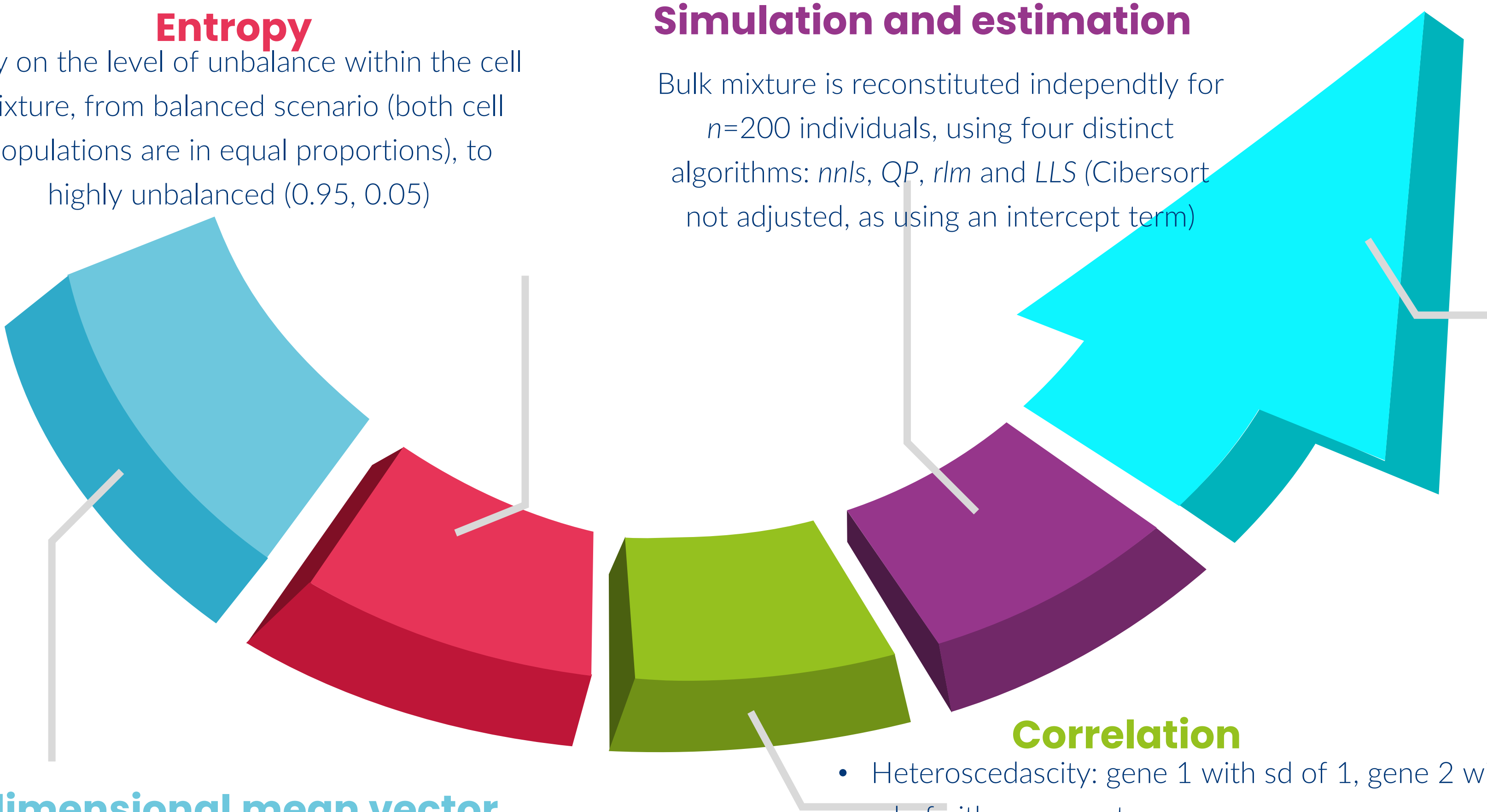
```
constrained_p <- .par2theta(p) # ensure the sum-to-one and non-negativity constraint
global_cov_matrix <- matrix(0, nrow = nrow(X), ncol = nrow(X),
                            dimnames = list(paste0("gene_", 1:nrow(X)),
                                            paste0("gene_", 1:nrow(X))))
for (j in 1:ncol(X)) {
  global_cov_matrix <- global_cov_matrix + constrained_p[j]^2*Sigma[,,j]
}
# deal with missing or infinite values when computing the covariance matrix
positive_definite <- FALSE
tryCatch({
  positive_definite <- is_positive_definite(global_cov_matrix)
},
error = function(e) {
  warning(paste("Error is:", e, "Global covariance matrix is not positive definite"))
})
if (positive_definite) {
  log_lik <- -log(det(global_cov_matrix)) - 1/2 * maha(y - X %*% constrained_p, global_cov_matrix) %>% as.numeric()
}
else {
  log_lik <- -Inf
}
```

$$p_1 = \frac{e^{\text{par}_1}}{e^{\text{par}_1} + e^{\text{par}_2} + 1} \quad p_2 = \frac{e^{\text{par}_2}}{e^{\text{par}_1} + e^{\text{par}_2} + 1} \quad p_3 = \frac{1}{e^{\text{par}_1} + e^{\text{par}_2} + 1}$$

$$\ell_{\mathbf{p}}(\mathbf{y}|\mathbf{X},\Sigma) = C + \log\left(\det\left(\left(\sum_{j=1}^{J} p_j^2\Sigma_j\right)^{-1}\right)\right) - \frac{1}{2}\underbrace{(\mathbf{y}-\mathbf{X}\mathbf{p})^{\top}\left(\sum_{j=1}^{J} p_j^2\Sigma_j\right)^{-1}(\mathbf{y}-\mathbf{X}\mathbf{p})}_{\text{squared Mahalanobis distance}}$$

the log-likelihood of the conditional distribution

Simulate the distribution: a toy example



Entropy

Play on the level of unbalance within the cell mixture, from balanced scenario (both cell populations are in equal proportions), to highly unbalanced (0.95, 0.05)

Simulation and estimation

Bulk mixture is reconstituted independtly for *n*=200 individuals, using four distinct algorithms: *nnls*, *QP*, *rlm* and *LLS* (Cibersort not adjusted, as using an intercept term)

Metrics

RMSE, MSE, (Pearson correlation) and adjusted R^2 coefficient are computed for each estimate of the ratios, for each individual

Box plots with mean values of the metrics scores are then computed, to study the impact of internal transcriptomic correlation genes on the the estimation

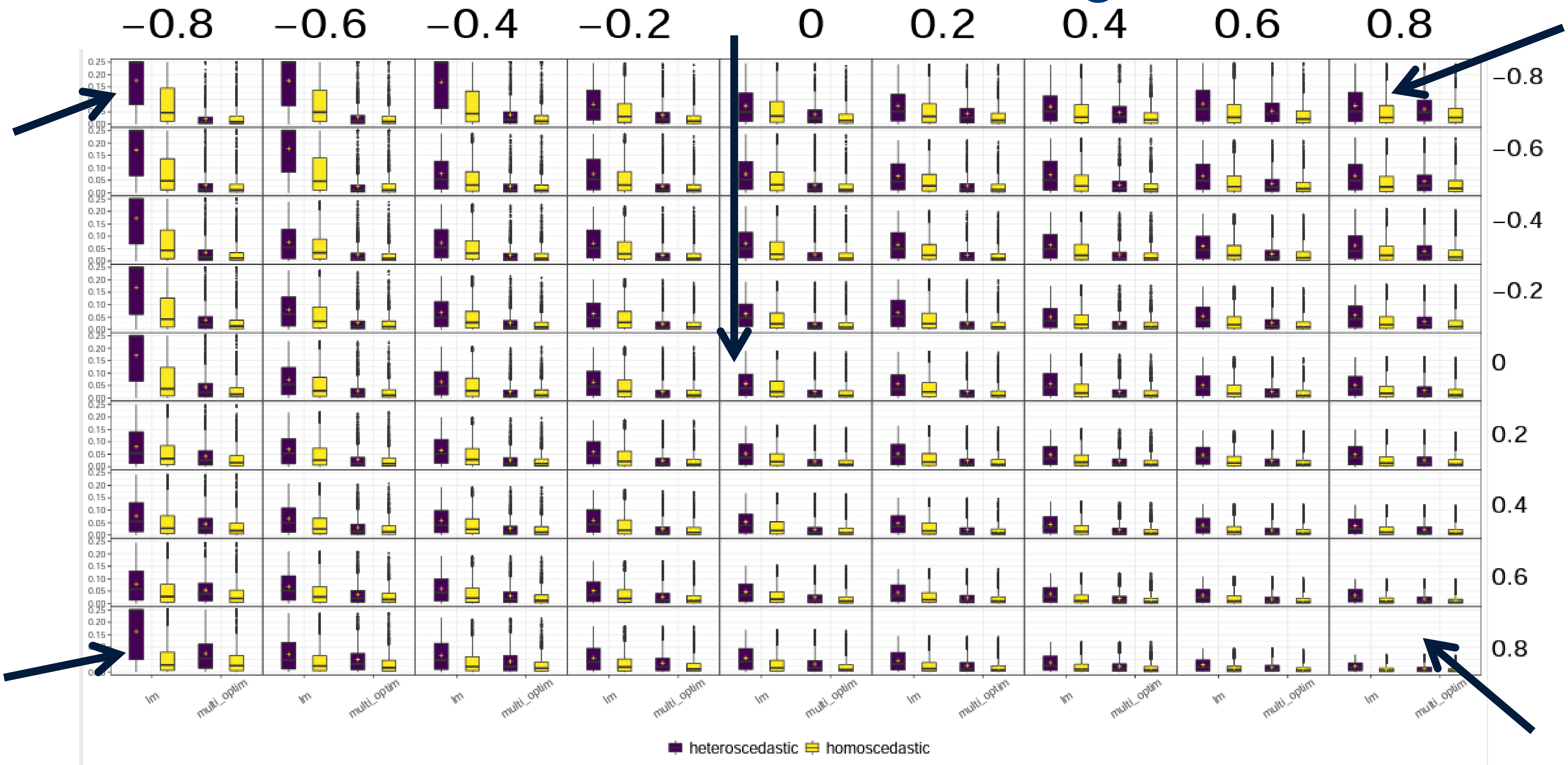
2-dimensional mean vector

A simple purified matrix, composed of two genes and two distinct cell populations

Correlation

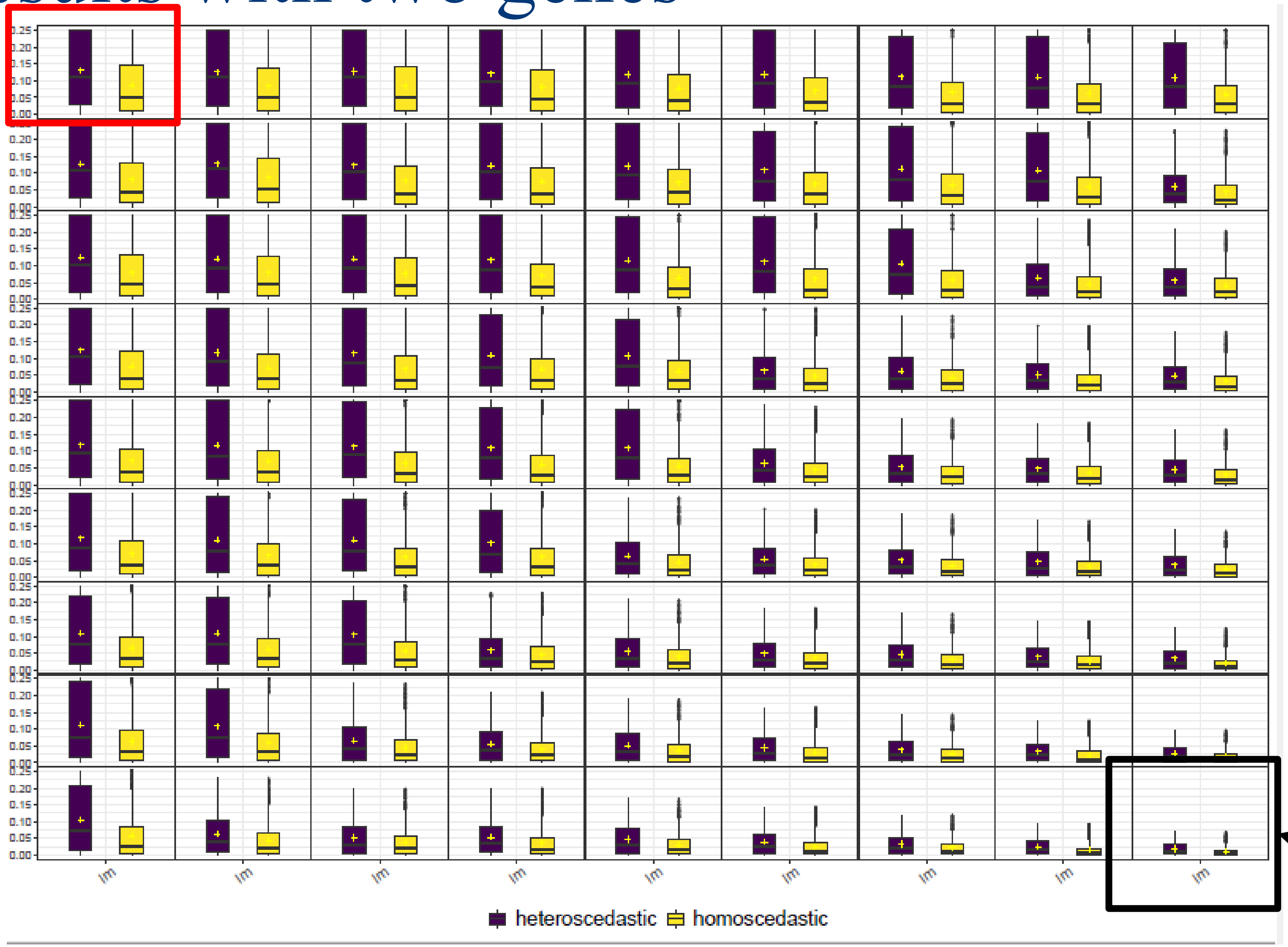
- Heteroscedascity: gene 1 with sd of 1, gene 2 with sd of either one or two
- All paired combinations of sequenced correlation levels between the two genes, played independently in the two populations

Simulation results with two genes



Corresponding boxplot representation of the MSE scores, comparing the performance of the two estimation algorithms

Simulation results with two genes

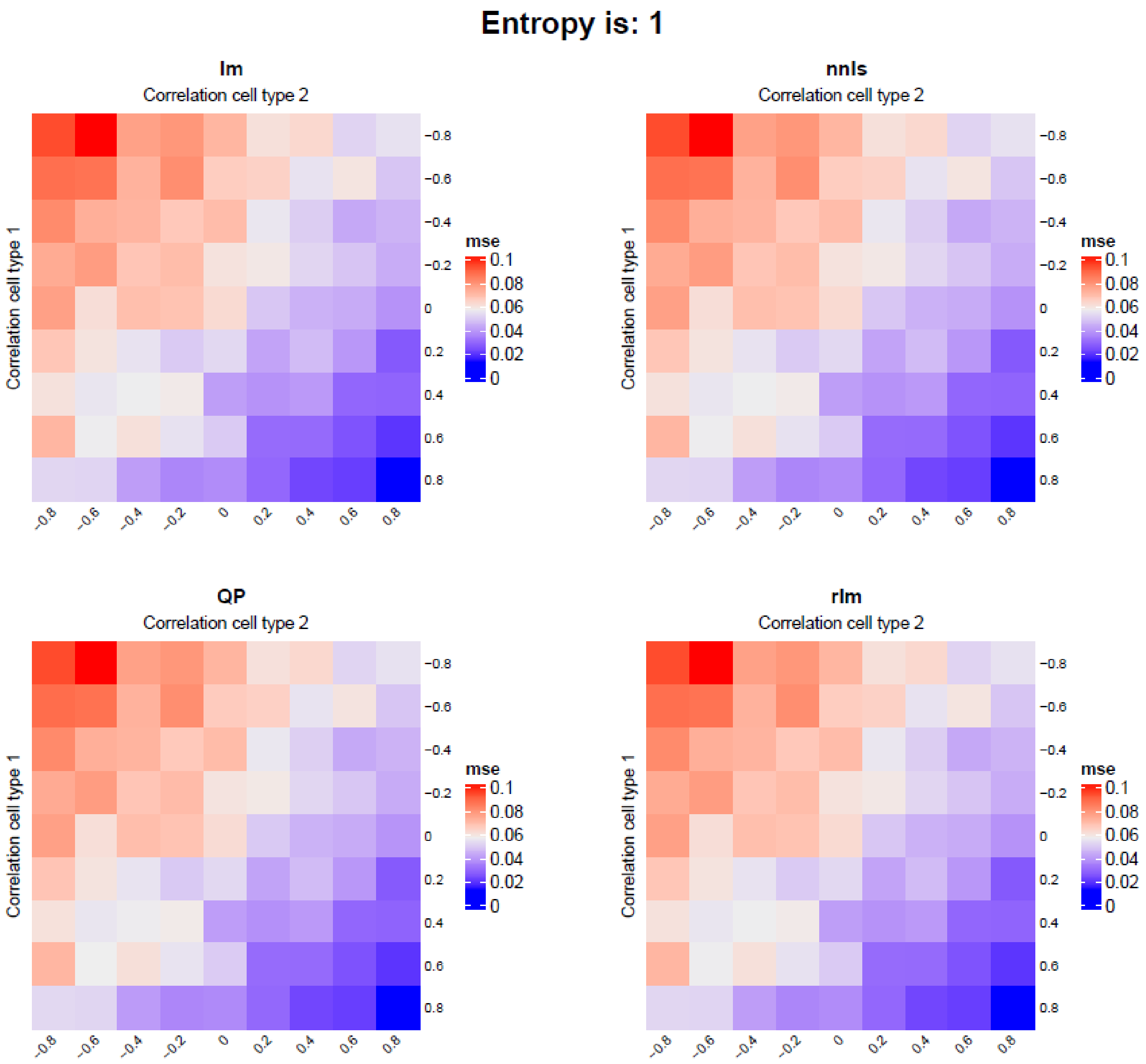


Worst estimation: genes in both populations are strongly negatively correlated

Best estimation: genes in both populations are strongly positively correlated

Same representation as before but highlighting the distribution of MSE term for each simulated scenario

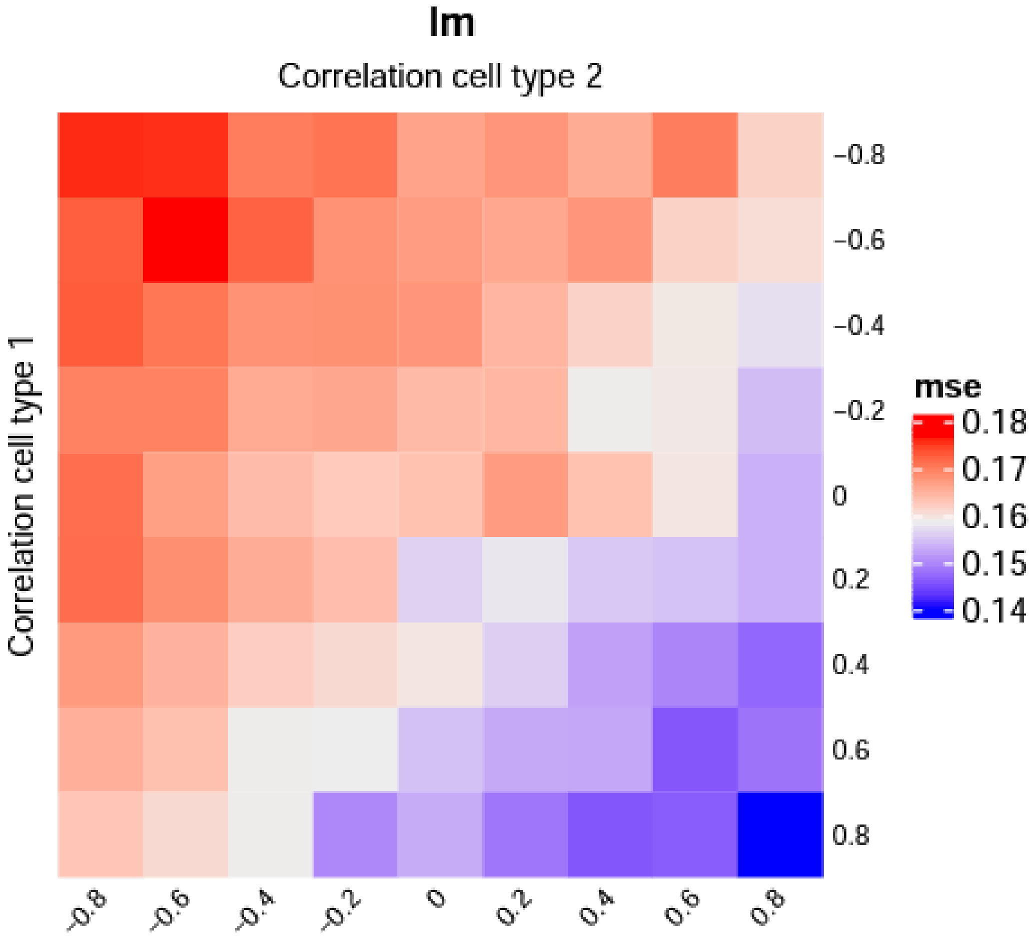
Simulation results with two genes

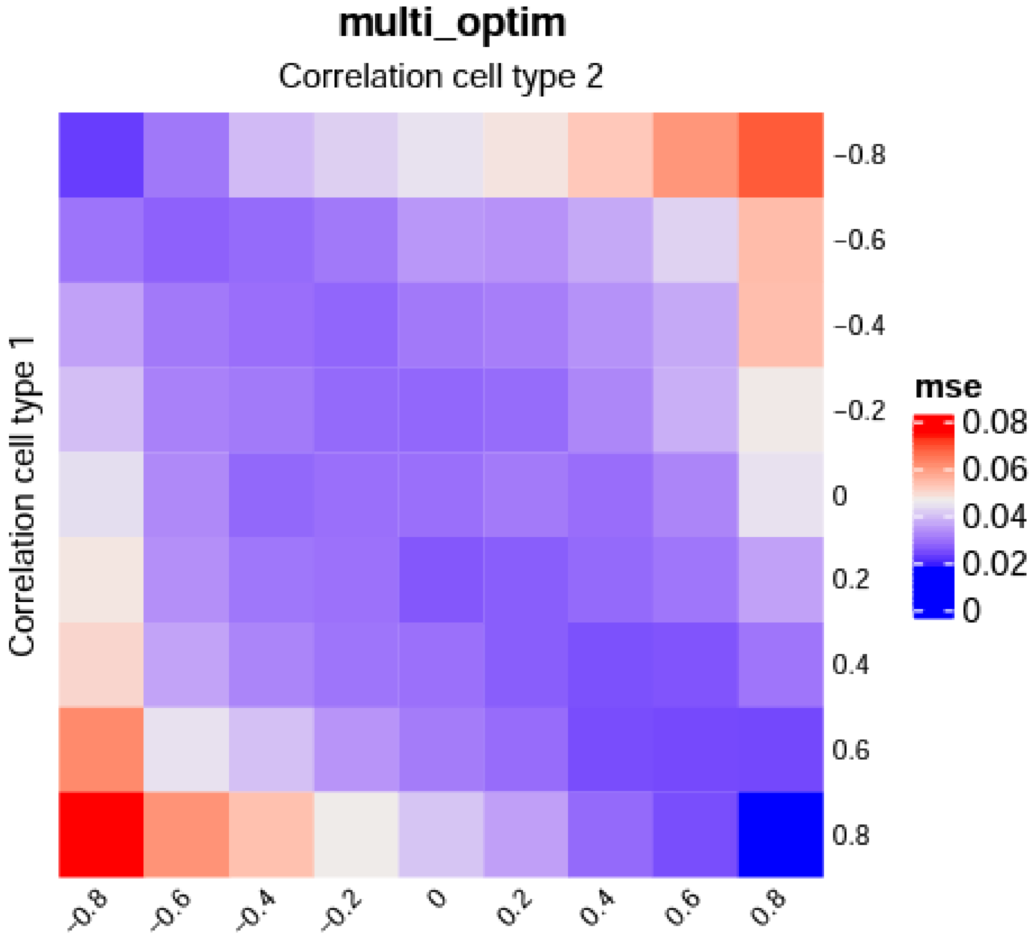


Heatmap of the MSE score,
with balanced proportions and highly overlapping genes

- One Heatmap per deconvolution method: with few input variables, same results returned
- Increasing entropy (disequilibrium between cell ratios) induces more bias, but same trend observable. Increasing heteroscedascity as well (both play on the overlap between the two multivariate population components)
- Worst scores in red (higher MSE), corresponding to the highest overlap between the two cell populations (in that scenario, when in pop cell 1, the two genes are strongly negatively correlated)
- Best scores obtained when the two genes are positively correlated, even better than in the scenario where no correlation is present (classical assumption of LS)
- Greater number of samples, to correctly re-build the multivariate distribution, than in the univariate case

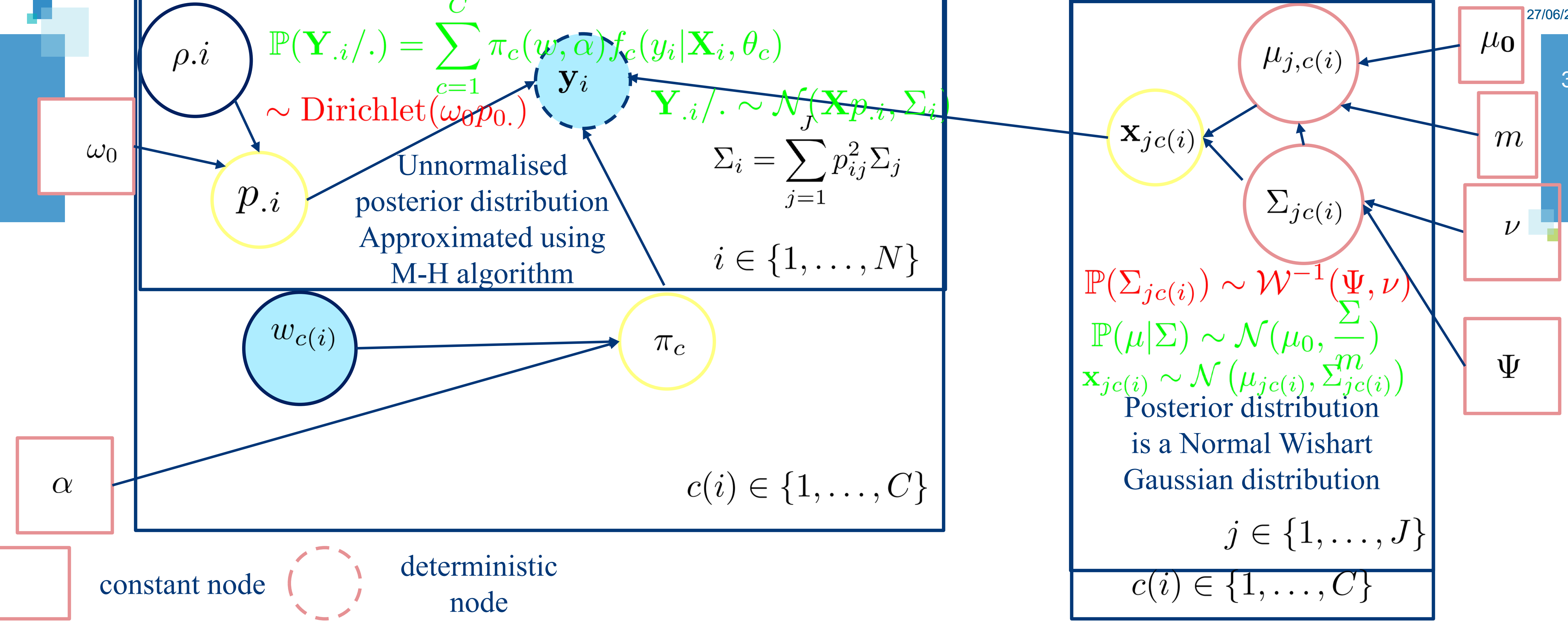
Simulation results with two genes





Same simulation parameters, but increasing the variance of gene 2

Respectively, estimation using the multivariate covariance information



$\mathbb{P}(\mathbf{Y}_{.i}/.) = \sum_{c=1}^{C} \pi_c(w,\alpha) f_c(y_i|\mathbf{X}_i,\theta_c)$

$\sim \text{Dirichlet}(\omega_0 p_{0.})$

$\mathbf{Y}_{.i}/. \sim \mathcal{N}(\mathbf{X}p_{.i},\Sigma_i)$

$\Sigma_i = \sum_{j=1}^{J} p_{ij}^2 \Sigma_j$

Unnormalised posterior distribution Approximated using M-H algorithm

$\mathbb{P}(\Sigma_{jc(i)}) \sim \mathcal{W}^{-1}(\Psi,\nu)$

$\mathbb{P}(\mu|\Sigma) \sim \mathcal{N}(\mu_0,\frac{\Sigma}{m})$

$\mathbf{x}_{jc(i)} \sim \mathcal{N}\left(\mu_{jc(i)},\Sigma_{jc(i)}\right)$

Posterior distribution is a Normal Wishart Gaussian distribution

□ constant node

○ deterministic node

○ stochastic node

● clamped node (observed)

Personal representation: multivariate distribution, accounting for phenotypical condition

Distribution probabilities

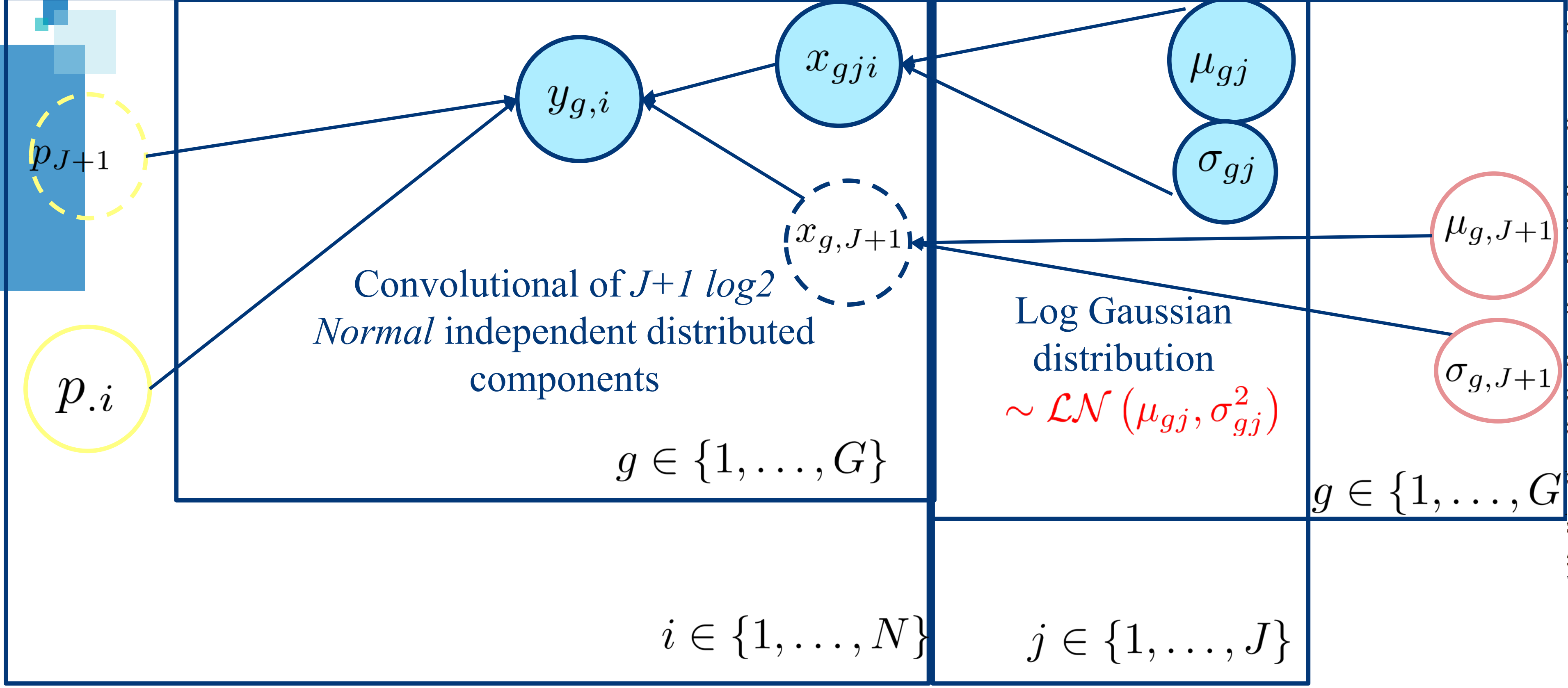
$f(\theta|\mathcal{D},\xi) \propto f(\mathcal{D}|\theta) f(\theta|\xi)$

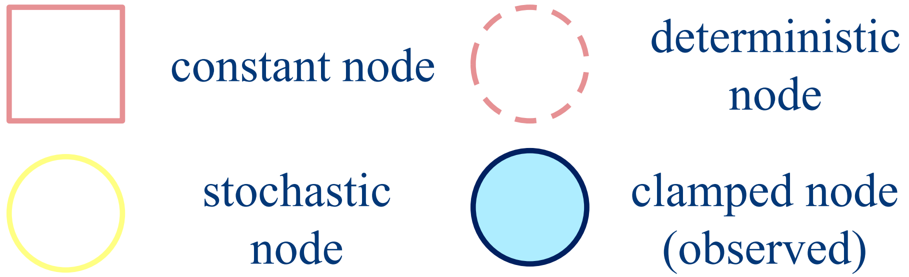
Prior laws	Likelihood Laws	Posterior Laws
$f(\theta\|\xi)$	$f(\mathcal{D}\|\theta)$	$f(\theta\|\mathcal{D},\xi)$

Parameters

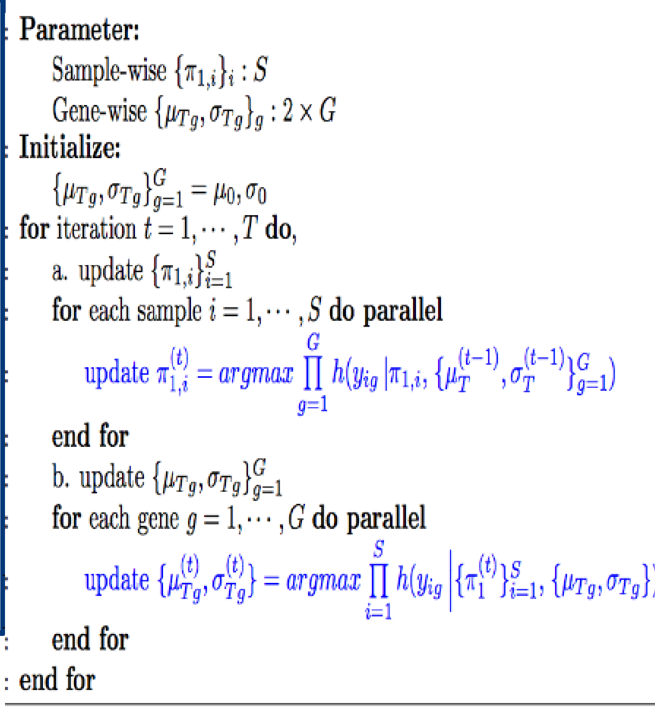
Plug-in parameters	Estimated parameters
$\tau = (\alpha,\rho,\omega_0,m,\nu,\mu_0,\Psi)$	$\theta = (p,\pi,\Sigma,\mu$

Probalistic model of DeMixt algorithm





Golden section search and parabolic interpolations (**numerical integration**) are used, as no closed form is available



```
: Parameter:
    Sample-wise $\{\pi_{1,i}\}_i : S$
    Gene-wise $\{\mu_{Tg}, \sigma_{Tg}\}_g : 2 \times G$
: Initialize:
    $\{\mu_{Tg}, \sigma_{Tg}\}_{g=1}^{G} = \mu_0, \sigma_0$
: for iteration $t = 1, \cdots, T$ do,
    a. update $\{\pi_{1,i}\}_{i=1}^{S}$
    for each sample $i = 1, \cdots, S$ do parallel
        update $\pi_{1,i}^{(t)} = argmax \prod_{g=1}^{G} h(y_{ig} | \pi_{1,i}, \{\mu_T^{(t-1)}, \sigma_T^{(t-1)}\}_{g=1}^{G})$
    end for
    b. update $\{\mu_{Tg}, \sigma_{Tg}\}_{g=1}^{G}$
    for each gene $g = 1, \cdots, G$ do parallel
        update $\{\mu_{Tg}^{(t)}, \sigma_{Tg}^{(t)}\} = argmax \prod_{i=1}^{S} h(y_{ig} \left| \{\pi_1^{(t)}\}_{i=1}^{S}, \{\mu_{Tg}, \sigma_{Tg}\}\right.$
    end for
: end for
```

ICM algorithm

$$Y = \log(X) \sim \text{Normal}$$

$$X = \exp(Y) \sim \text{LogNormal}$$

Distribution probabilities

Likelihood Laws

$$f(\mathcal{D}|\theta)$$

Parameters

Estimated parameters

$$\theta = (p, \mu, \sigma, \mathbf{X})$$

SERVIER

Conclusion

Main innovations in our new deconvolution algorithm

Data collection

Poorly described cell populations, full exploitation of Encode and Blueprint datasets

Automatic annotation and description of cellular ontology

Curation

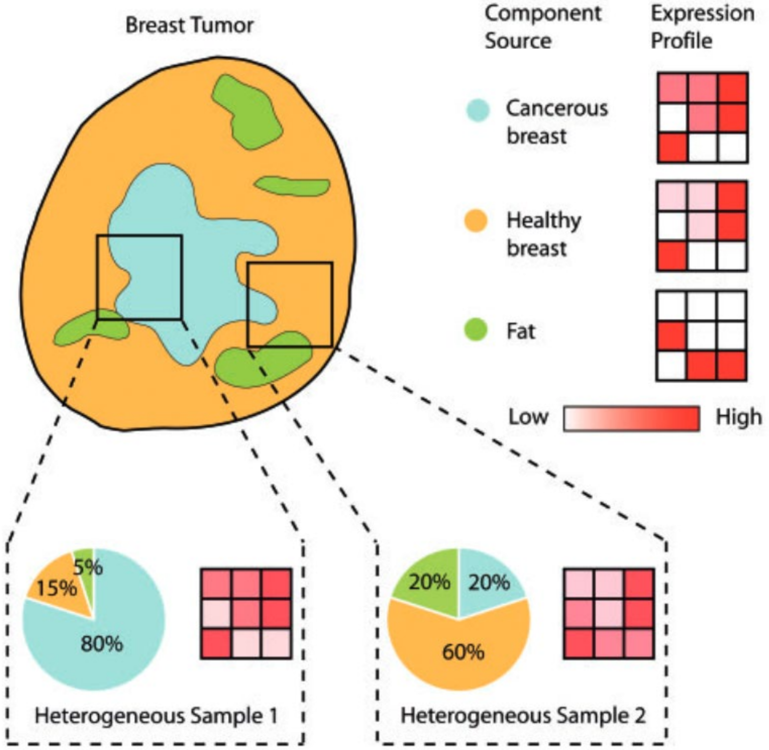
Refine selection of relevant genes:

- Automated method for discarding background noise
- Innovative feature-selection algorithms, using both the differential expression and the covariance structure

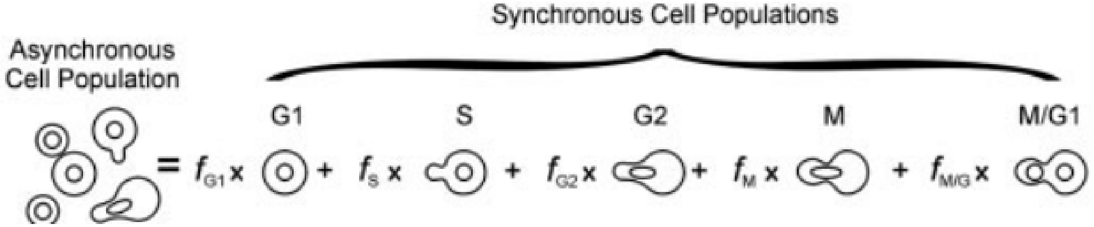
Connectivity

Algorithm closer to biological models, accounting for the co-transcriptomic expression between the genes of the purified cell populations

The complexity of the biological medium

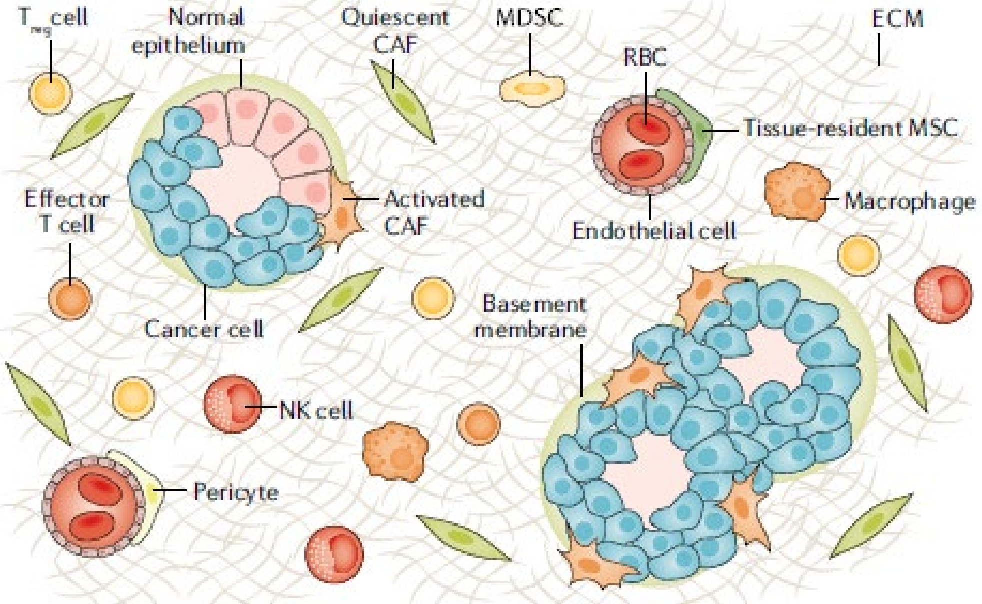


Mixture of tissues
Quon and Morris, 2009


Synchronous Cell Populations
Asynchronous Cell Population
G1
S
G2
M
M/G1
= f_{G1} x + f_S x + f_{G2} x + f_M x + $f_{M/G}$ x

Mixture of cell phases

Lu et al, 2003



Mixture of cell populations
Finotello and Trajanoski 2018

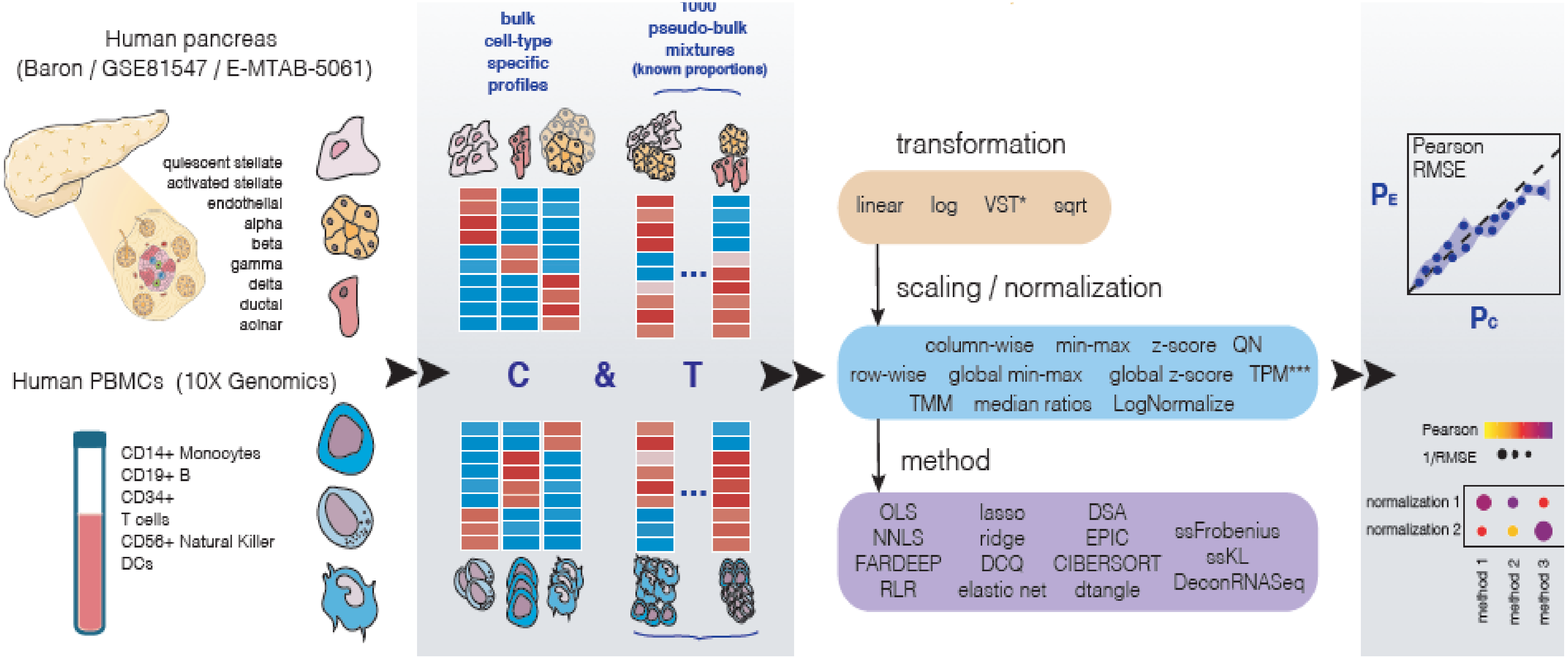
Canonical deconvolution pipeline

Step 1: collection and curation of datasets

Step 2: learn for each cell-type its associated transcriptomic network structure

Step 3: innovative deconvolution algorithm, taking profit of the transcripts interactions

Step 4: biological and statistical evaluation



Fa et al. 2020

Computational pipeline for the estimation of the ratios using transcriptomic data

Framework of the multivariate probalistic model

independence between the cell types + *invariant* property of Gaussian distributions under *affine* transformation

the *conditional distribution* of the bulk mixture follows a *multivariate Gaussian distribution*

$$\mathbf{y}_i/\mathbf{X} \sim \mathcal{N}_G(\mu_i, \Sigma_i)$$

$$\mu_i = \mathbf{X}p_i$$

mean matrix

$$\Sigma_i = \sum_{j=1}^{J} p_{ij}^2 \Sigma_j$$

covariance matrix

Step 1: ***X*** is drawn independantly from a multivariate Gaussian distribution for each cell type

$$\mathbf{X}_j \sim \mathcal{N}_G\left(\hat{\mu_j}, \hat{\mathbf{\Sigma}}_j\right)$$

Step 2: Reconstitute *Y*, the bulk mixture, by summing the weighted contribution of each cellular expression profile.

$$\mathbf{y}_i = \mathbf{X}p_i$$

matrix form

$$y_{gi} = \sum_{j=1}^{J} x_{gj}p_{ji}$$

algebraic form