Bayesian Inference model with Nested effects to perform Differential Gene Expression analysis from Multi-level Spatial Transcriptomics data with Multiple Conditions

Lira Pi, Matthew Ryals, Jialie Luo, and Jake Gagnon

October 26, 2023

Scientific Journey on Omics

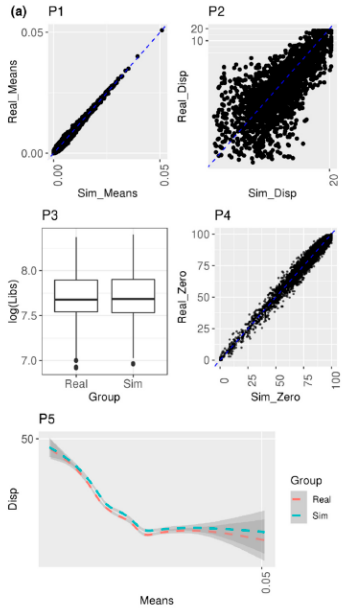
scRNA-seq

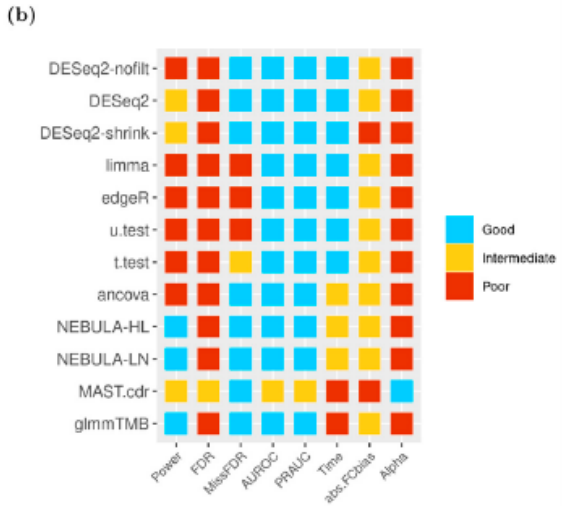
life MDPI

Article

Recommendations of scRNA-seq Differential Gene Expression Analysis Based on Comprehensive Benchmarking

Jake Gagnon [1], Lira Pi [2], Matthew Ryals [2], Qingwen Wan [2], Wenxing Hu [3], Zhengyu Ouyang [4], Baohong Zhang [3,*] and Kejie Li [3,*]





scRNA-seq + scATAC-seq

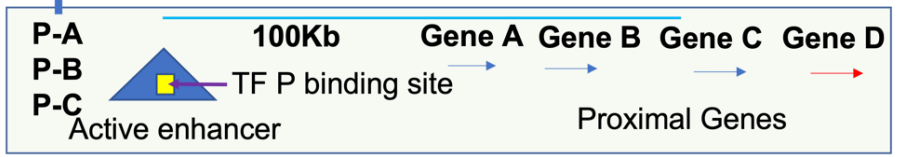
Highly-connected elements of the zebrafish enveloping layer transcriptional regulatory network are enriched for orofacial cleft risk genes

Sunil K Singh[1,2], Annika Helverson[3], Colin Kenny[1,2,4], Kaylia M Duncan[1,5], Lira Pi[1,6], Edward B Li[7], Sarah Curtis[8], Eric Liao,[5,7], Elizabeth Leslie[8], Patrick Breheny[3], Robert A Cornell[1,2]

Shield Stage multiome
RNA expression in EVL cells → **Target Genes**
3,787 Genes x 394 Samples

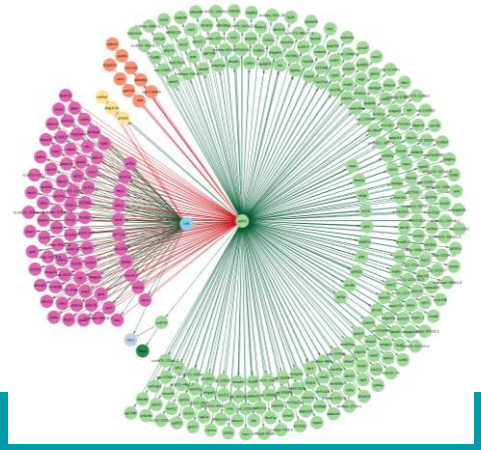
Prior Information
ATAC-seq data from multiome → **Prior Matrix**
430 TFs x 3,787 Genes

→ **Periderm TRN**
gLASSO

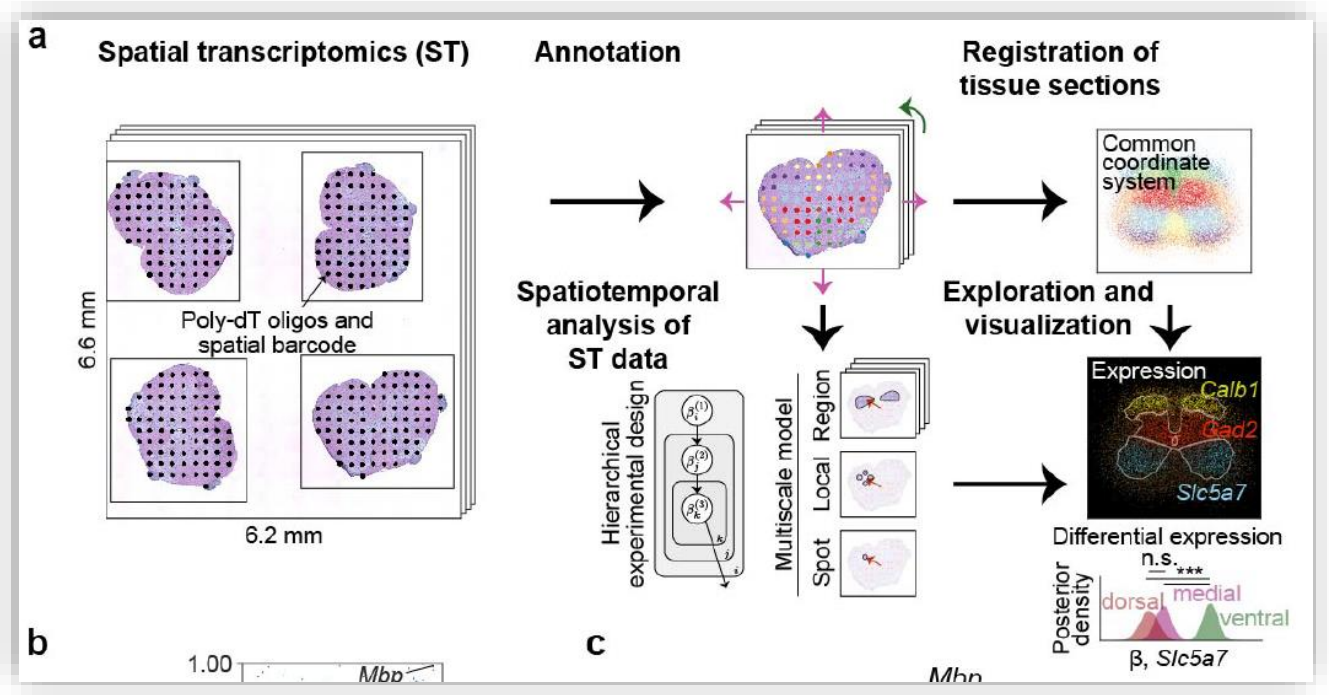


- Graphical Lasso (gLASSO)

$$\log det\ \Theta - tr(S\Theta) - ||\Theta * P||_1$$



Spatial Transcriptomics Data



- a. illustration of the proposed ST analysis workflow: experimental design of statistical spatio-temporal data analysis *(source: Aijo et al. bioRxiv 2019)*

- With spatial transcriptomics,
  - Structure: we can characterize tissue organization and architecture at the **single-cell** or subcellular resolution level
  - Quantification of expression: we can quantify the **expression level of individual genes**
  - Interaction: it is possible to obtain information on the transcriptomes of a **single cell** or a small group of cells, while maintaining the information of **where the cell (or group of cells) is located within the tissue**. Enabling us to understand how and why a specific cell or small group of cells respond to the surrounding environment.
    - Ligand-receptor interaction between neighboring cells.
    - Signaling pathways between neighboring cells.
    - **DEG between multiple conditions per location**

Splotch and its Limitations

Title

Splotch: Robust estimation of aligned spatial temporal gene expression data.

Authors

Tarmo Äijö*, Silas Maniatis*, Sanja Vickovic*, Kristy Kang, Miguel Cuevas, Catherine Braine,

Hemali Phatnani, Joakim Lundeberg, Richard Bonneau†

- Splotch was the most relevant method to resolve DEG discovery.

First let us compare the expression of *Gfap* in ventral horn between WT P120 and G93A P120

In [9]:

```
# define the gene of interest
gene = 'Gfap'

# make sure we have analyzed it
assert gene in samples, 'Error: %s not found!'%(gene)

# define the level of interest (WT P120 and G93A P120 are level 1)
level = 'beta_level_1'
# define the variables of interest from that level
beta_variables_of_interest = ['WT p120','G93A p120']
# define the aar of interest
aar_variable_of_interest = 'Vent_Horn'

# find the mappings from names to indices (Stan has no dictionaries)
beta_variable_indices = to_stan_variables(beta_mapping[level],beta_variables_of_interest)
aar_index = to_stan_variables(aar_names,aar_variable_of_interest)
```

Calculate the Savage-Dickey density ratio to quantify the difference

In [10]:

```
print("Approximated Bayes factor (BF) is %.4f"%(
    savagedickey(samples[gene][level][:,beta_variable_indices[0],aar_index].flatten(),
                 samples[gene][level][:,beta_variable_indices[1],aar_index].flatten())))
```

```
Approximated Bayes factor (BF) is 408.7260
```

https://github.com/tare/Splotch/blob/master/Tutorial.ipynb

- Running the Python-Stan codes for one single gene to test gene-expression differentiation between two groups given specific region took several hours even in HPC!!
- Splotch was developed on ST array design, but our data was 10x Visium.

Multi-level (Hierarchical) Spatial Design Using R-INLA

- The zero-inflated Poisson model is expressed as

$$y_{i,j,k} \big| (s_{j,k}, \lambda_{i,j,k}, \theta_i^p) \sim ZIP(s_{j,k}\lambda_{i,j,k}, \theta_i^p)$$

Where $y_{i,j,k}$ = the number of UMIs for i^{th} gene at k^{th} spot on j^{th} tissue section; $s_{j,k}$ = size (scaling) factor; $\lambda_{i,j,k}$ = rate parameter; θ_i^p= zero-inflation parameter

$$\exp(\lambda_{i,j,k}) = X_{j,k}^T \beta_{i,g} + B_{b:g} + T_{t:(b:g)} + \psi_{i,j,k} + \varepsilon_{i,j,k}$$

Where $X_{j,k}$ = binary indicator of treatment groups; index $b:g$ denotes that biological samples (mice) are nested within a treatment group; index $t:(b:g)$ denotes nesting of technical samples (multiple tissue sections) within mice.

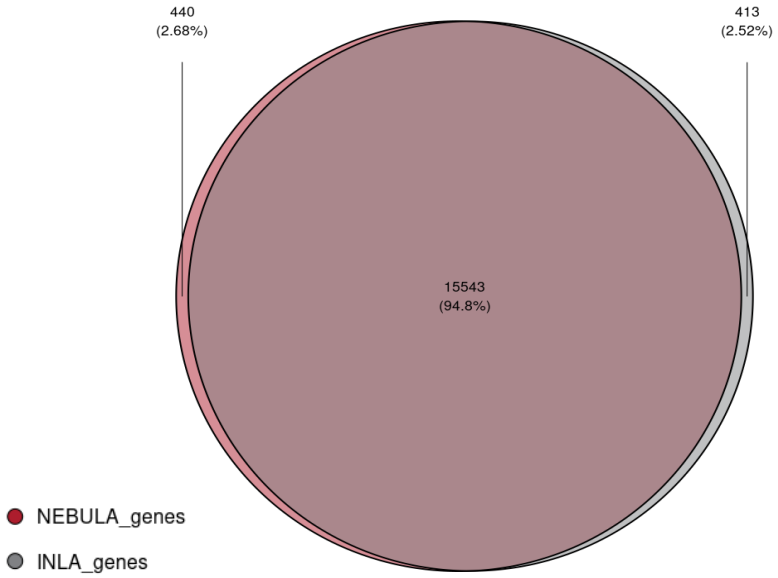
```
#Priors
# half normal for level.2 and level.3 variations
sigma.prior.sex_mouse = "expression:
tau0 = 1;
sigma  =exp(-theta/2);
log_dens = log(2) - 0.5*log(2*pi) + 0.5*log(tau0);
log_dens = log_dens - 0.5*tau0*sigma^2;
log_dens = log_dens - log(2) - theta/2;
return (log_dens);
"
# half-normal for spot-level variation
sigma.prior.epsilon = "expression:
tau0 = 100/9;  ##LP
sigma  =exp(-theta/2);
log_dens = log(2) - 0.5*log(2*pi) + 0.5*log(tau0);
log_dens = log_dens - 0.5*tau0*sigma^2;
log_dens = log_dens - log(2) - theta/2;
return (log_dens);
"
# zero-inflation prior:
theta_prior <- list(theta = list(prior = "logitbeta", param=c(2,1)))
# CAR prior for log(precision parameter) ## LP
x <- seq(0, 1500, by = 0.1)
log_dens <- dinvgamma(x, shape = 1, rate = 1, log = TRUE)
invgamma_prior <- paste0("table: ",
                         paste(c(x, log_dens), collapse = " ")
                         )
```

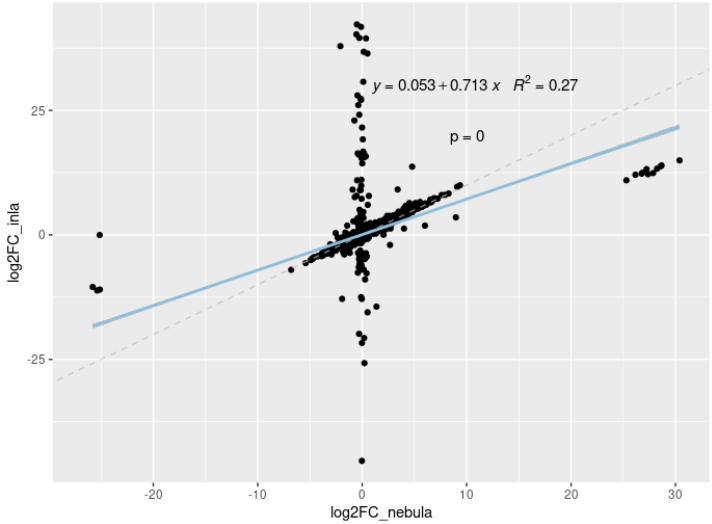
- Conditional Autoregressive (CAR)

$$\psi_{i,j} | a_i, \tau_i, \boldsymbol{W}_j \sim N\left(\boldsymbol{0}, (\tau_i \boldsymbol{D}_i (\boldsymbol{I} - a_i \boldsymbol{D}_j^{-1} \boldsymbol{W}_j))^{-1}\right),$$

```
inla_res.FM <- tryCatch({inla(count ~ 0 + Level.1 +
                              f(ID2, model = "z", Z = Zlevel.2, hyper = sigma.prior.sex_mouse)+
                              f(ID3, model = "z",Z = Zlevel.3,hyper = sigma.prior.sex_mouse) +
                              f(ID, model = "generic1", Cmatrix = C, hyper = invgamma_prior)+  ##LP
                              f(epsilon, model = "iid", hyper = sigma.prior.epsilon),
                            E=size_factors_vec[inla_index],
                            control.fixed=list(
                              mean = list("Level.1" = 0),
                              prec = list("Level.1" = 0.25)
                            ),
                            data = inla_data, control.family= list(hyper = theta_prior),num.threads=16,
                            family = "zeroinflatedpoisson1",control.compute=list(config = TRUE))}, error=function(e) "FM failed!")
```

Results from Real Data





- 16.5k genes were tested by two DE (differential expression) methods – doable by INLA, but Stan
  - The different sets of DE testing probably came from different filtering application of CPC > 0.005 before/after sub-setting contrast groups.
- Degree of concordance between two DE methods in terms of log2FC estimates across full set of tested genes.
  - NEBULA-HL estimates some log2FC close to zero for which INLA has estimated very large log2FC value.

Key References

- Gagnon J, **Pi L**, Ryals M, Wan Q, Hu W, Ouyang Z, Zhang B, Li K. Recommendations of scRNA-seq Differential Gene Expression Analysis Based on Comprehensive Benchmarking. Life (Basel). 2022 Jun 7;12(6):850. doi: 10.3390/life12060850. PMID: 35743881; PMCID: PMC9225332.

- Gómez-Rubio, Virgilio (2020). Bayesian Inference with INLA. Chapman & Hall/CRC Press. Boca Raton, FL.

- Tarmo Äijö, Silas Maniatis, Sanja Vickovic, Kristy Kang, Miguel Cuevas, Catherine Braine, Hemali Phatnani, Joakim Lundeberg, Richard Bonneau. Splotch: Robust estimation of aligned spatial temporal gene expression data. bioRxiv 757096; doi: https://doi.org/10.1101/757096

Appendix: Bayes Factor

```
#Reduced model with Level.1 removed
inla_res.RM <- tryCatch({inla(count ~ 0 +
                                f(ID2, model = "z", Z = Zlevel.2, hyper = sigma.prior.sex_mouse)+
                                f(ID3, model = "z",Z = Zlevel.3,hyper = sigma.prior.sex_mouse) +
                                f(ID, model = "generic1", Cmatrix = C, hyper = invgamma_prior)+  ##LP
                                f(epsilon, model = "iid", hyper = sigma.prior.epsilon),
                              E=size_factors_vec[inla_index],
                              control.fixed=list(
                                mean = list("Level.1" = 0),
                                prec = list("Level.1" = 0.25)
                              ),
                              data = inla_data, control.family= list(hyper = theta_prior),num.threads=16,
                              family = "zeroinflatedpoisson1",control.compute=list(config = TRUE))}, error=function(e) "RM failed!")

#Calculate BF MLIK full model MLIK minus reduced model MLIK
## Bayes Factor: LP
BF_list <- tryCatch({exp(inla_res.FM$mlik[rownames(inla_res.FM$mlik)=="log marginal-likelihood (Gaussian)",1]-
                                inla_res.RM$mlik[rownames(inla_res.RM$mlik)=="log marginal-likelihood (Gaussian)",1])},
                        error=function(e) "Model failed!")
```

Finally, the marginal likelihood can be use to compute Bayes factors (Gelman et al. 2013) to compare two given models. The Bayes factor for models $\mathcal{M}_1$ and model $\mathcal{M}_2$ is given by

$$\frac{\pi(\mathcal{M}_1 \mid \mathbf{y})}{\pi(\mathcal{M}_2 \mid \mathbf{y})} = \frac{\pi(\mathbf{y} \mid \mathcal{M}_1)\pi(\mathcal{M}_1)}{\pi(\mathbf{y} \mid \mathcal{M}_2)\pi(\mathcal{M}_2)}$$