RNA-seq co-expression analysis using mixture models

A. Rau, C. Maugis-Rabusseau, M.-L. Martin-Magniette, G. Celeux

September 29, 2015

Netbio @ Paris

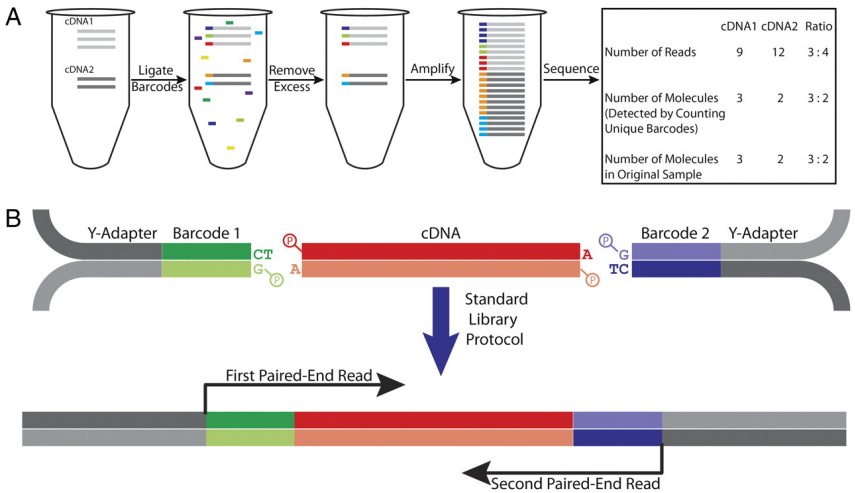
Gene (co-)expression

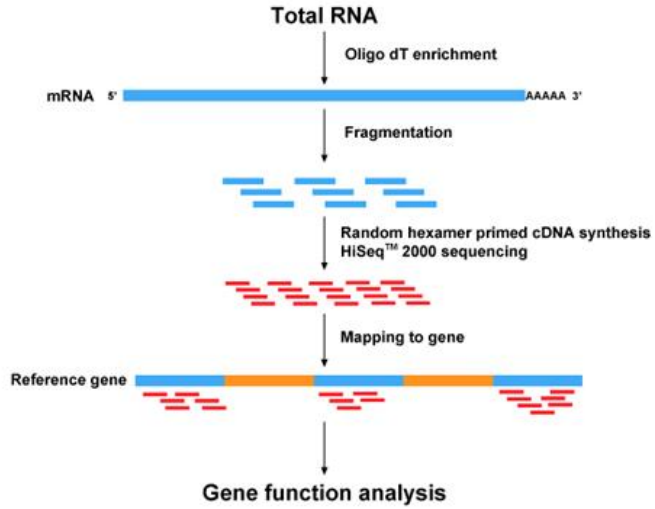
- Transcriptome data: main source of 'omic information available for living organisms
  - Microarrays (∼1995 -)
  - High-throughput sequencing (HTS): RNA-seq (∼2008 -)
- Comparison of two conditions (hypothesis tests) → Differential expression analysis

Co-expression (clustering) analysis

- Study gene expression behavior across several conditions
- Co-expressed genes may be involved in similar biological process(es)
  ⇒ study genes without known or predicted function (orphan genes)

High-throughput transcriptome sequencing data (RNA-seq)



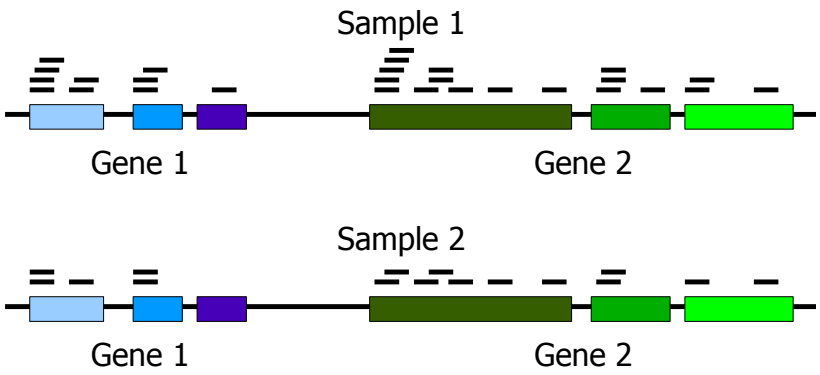


- Reads aligned or directly mapped to the genome to get counts per genomic feature (discrete data) $\Rightarrow$ digital measures of gene expression

RNA-seq data, continued

Some statistical challenges of RNA-seq data analysis

- Discrete, non-negative, and skewed data with very large dynamic range (up to 5+ orders of magnitude)
- Sequencing depth (= "**library size**") varies among experiments, and other technical biases...
- Counts correlated with gene length



Gene	E1	E2	E3
13CDNA73	4	0	6
A2BP1	19	18	20
A2M	2724	2209	13
A4GALT	0	0	48
AAAS	57	29	224
AACS	1904	129	4
AADACL1	3	13	239
[...]			

To date, most methodological developments are for experimental design, normalization, and differential analysis...

Some notation

Notation

Let $Y_{ij\ell}$ be the count (expression measure) for gene i in replicate ℓ of condition j, with corresponding observed value $y_{ij\ell}$.

- Let $s_{j\ell}$ be the **library size** in replicate ℓ of condition j
- Let $\mathbf{y} = (y_{ij\ell})$ be the $n \times \sum_j L_j$ matrix of counts for all genes and variables and $\mathbf{y}_i$ the ith row of the matrix

Finite mixture models

Model-based clustering

- Rigourous framework for parameter estimation and model selection
- **Output**: each gene assigned a probability of cluster membership

Assume data $\mathbf{y}$ come from K distinct subpopulations, each modeled separately:

$$f(\mathbf{y}|K, \boldsymbol{\Psi}_K) = \prod_{i=1}^{n} \sum_{k=1}^{K} \pi_k f_k(\mathbf{y}_i|\boldsymbol{\theta}_k)$$

- $\boldsymbol{\Psi}_K = (\pi_1, \ldots, \pi_{K-1}, \boldsymbol{\theta}')'$
- $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_K)'$ are the mixing proportions, where $\sum_{k=1}^{K} \pi_k = 1$

Finite mixture models for RNA-seq data

$$f(\mathbf{y}|K, \boldsymbol{\Psi}_K) = \prod_{i=1}^{n} \sum_{k=1}^{K} \pi_k f_k(\mathbf{y}_i|\boldsymbol{\theta}_k)$$

- For microarray data, we often assume $\mathbf{y}_i|k \sim \text{MVN}(\boldsymbol{\mu}_k, \Sigma_k)$...

Finite mixture models for RNA-seq data

$$f(\mathbf{y}|K, \boldsymbol{\Psi}_K) = \prod_{i=1}^{n} \sum_{k=1}^{K} \pi_k f_k(\mathbf{y}_i|\boldsymbol{\theta}_k)$$

- For microarray data, we often assume $\mathbf{y}_i|k \sim \text{MVN}(\boldsymbol{\mu}_k, \Sigma_k)$...
- For RNA-seq data, we need to choose the family and parameterization of $f_k(\cdot)$. One possibility:

$$\mathbf{y}_i|k \sim \prod_{j=1}^{J} \prod_{\ell=1}^{L_j} \mathcal{P}(y_{ij\ell}|\mu_{ij\ell k})$$

Finite mixture models for RNA-seq data

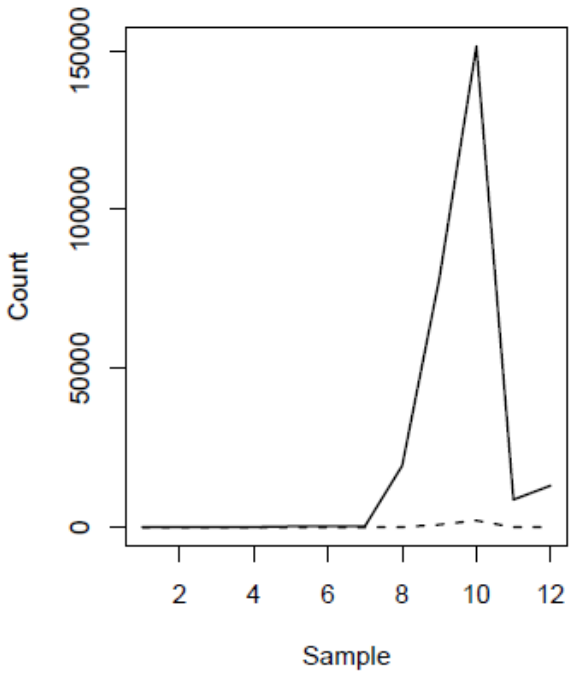
$$f(\mathbf{y}|K, \boldsymbol{\Psi}_K) = \prod_{i=1}^{n} \sum_{k=1}^{K} \pi_k f_k(\mathbf{y}_i|\boldsymbol{\theta}_k)$$

- For microarray data, we often assume $\mathbf{y}_i|k \sim \text{MVN}(\boldsymbol{\mu}_k, \Sigma_k)$...
- For RNA-seq data, we need to choose the family and parameterization of $f_k(\cdot)$. One possibility:

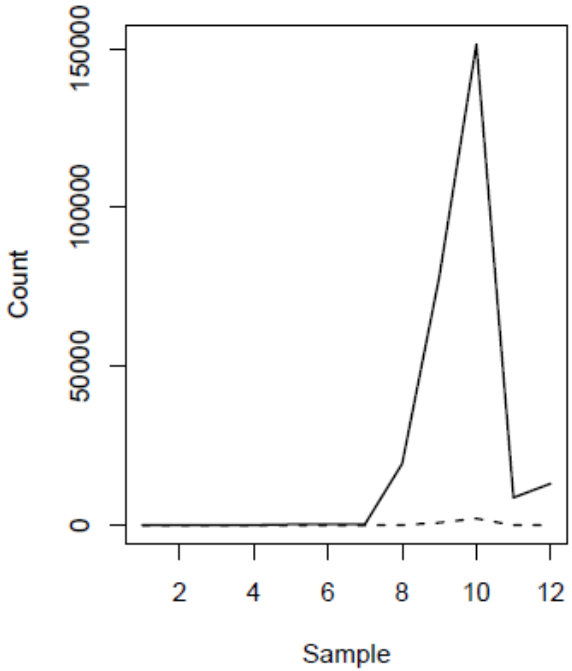
$$\mathbf{y}_i|k \sim \prod_{j=1}^{J} \prod_{\ell=1}^{L_j} \mathcal{P}(y_{ij\ell}|\mu_{ij\ell k})$$

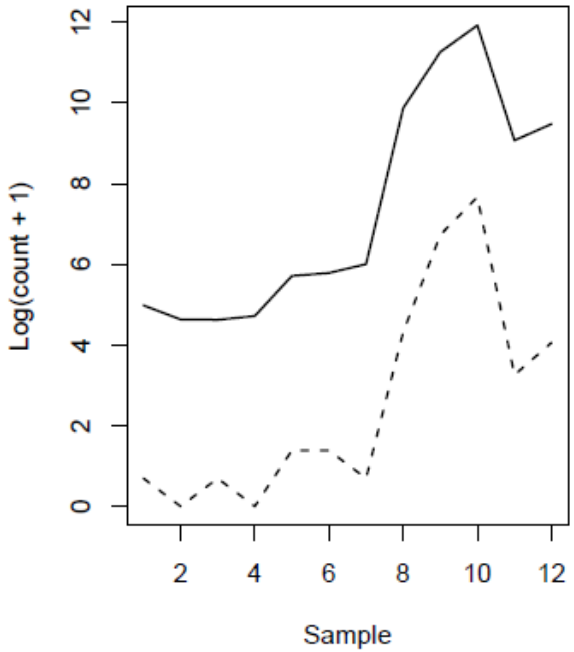
Question: How to parameterize the mean $\mu_{ij\ell k}$ to obtain meaningful clusters of co-expressed genes?

Which genes should be clustered?

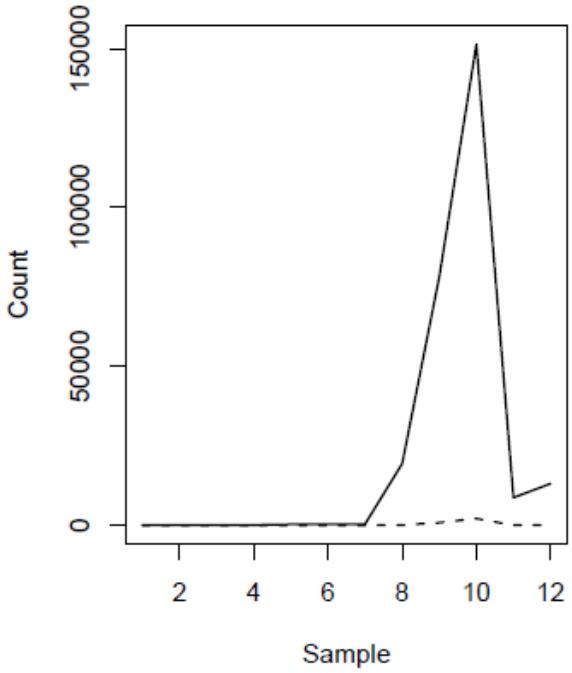


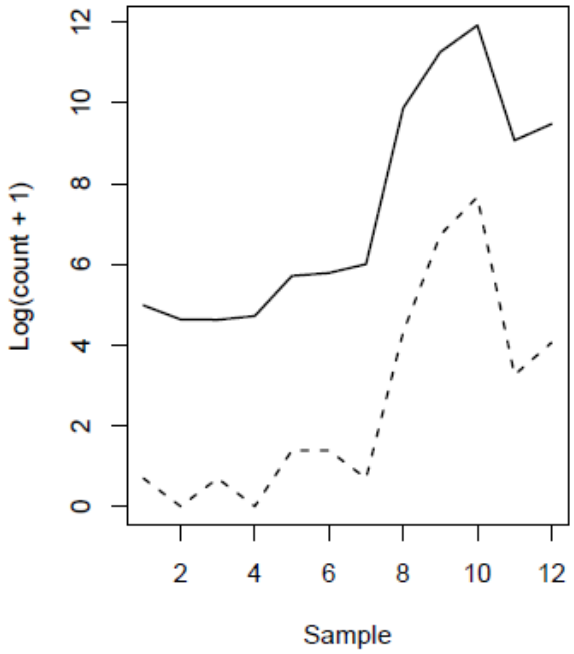
Which genes should be clustered?

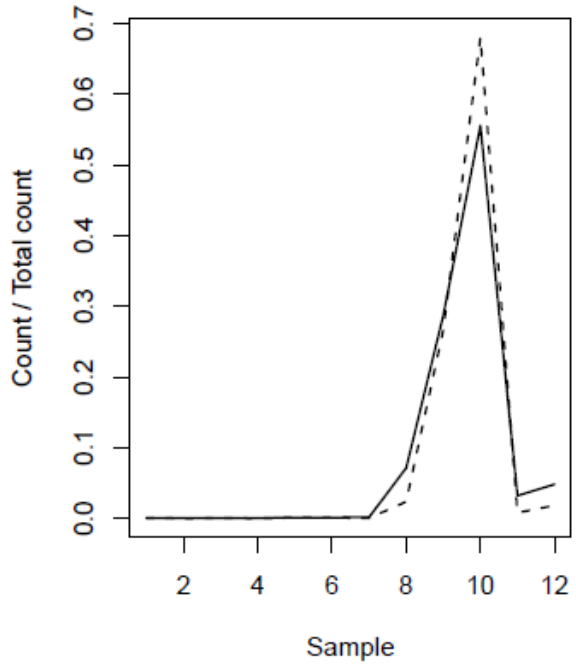




Which genes should be clustered?







Poisson mixture model for RNA-seq data

Consider $y_{ij\ell}|k \sim \text{Poisson}(y_{ij\ell}|\mu_{ij\ell k})$, where

$$\mu_{ij\ell k} = w_i s_{j\ell} \lambda_{jk}$$

- w_i : overall expression level of gene i $(= y_{i..})$
- $s_{j\ell}$: normalized library size[a]
- $\boldsymbol{\lambda}_k = (\lambda_{jk})$: parameters that define profiles of genes in each cluster[b]

[a] Estimated from data using standard techniques and considered to be fixed

[b] For identifiability of model, we assume $\sum_{j,\ell} \lambda_{jk} s_{j\ell} = 1$ for all k

Poisson mixture model for RNA-seq data

Consider $y_{ij\ell}|k \sim \text{Poisson}(y_{ij\ell}|\mu_{ij\ell k})$, where

$$\mu_{ij\ell k} = w_i s_{j\ell} \lambda_{jk}$$

- w_i : overall expression level of gene i $(= y_{i..})$
- $s_{j\ell}$: normalized library size[a]
- $\boldsymbol{\lambda}_k = (\lambda_{jk})$: parameters that define profiles of genes in each cluster[b]

[a] Estimated from data using standard techniques and considered to be fixed
[b] For identifiability of model, we assume $\sum_{j,\ell} \lambda_{jk} s_{j\ell} = 1$ for all k

- Genes assigned to the same cluster if they share the same **profile of variation** around their mean count across all conditions

Parameter estimation

The log likelihood is

$$L(\boldsymbol{\Psi}_K|\mathbf{y}, K) = \log\left[\prod_{i=1}^{n} f(\mathbf{y}_i|K, \boldsymbol{\Psi}_K)\right] = \sum_{i=1}^{n} \log\left[\sum_{k=1}^{K} \pi_k f(\mathbf{y}_i|\boldsymbol{\theta}_k)\right],$$

where $\boldsymbol{\theta}_k = (w_i, \lambda_{1k}, \ldots, \lambda_{dK})'$

Parameter estimation

The log likelihood is

$$L(\boldsymbol{\Psi}_K|\mathbf{y}, K) = \log\left[\prod_{i=1}^{n} f(\mathbf{y}_i|K, \boldsymbol{\Psi}_K)\right] = \sum_{i=1}^{n} \log\left[\sum_{k=1}^{K} \pi_k f(\mathbf{y}_i|\boldsymbol{\theta}_k)\right],$$

where $\boldsymbol{\theta}_k = (w_i, \lambda_{1k}, \ldots, \lambda_{dK})'$

- **Estimation approach (EM):** mixture parameters are estimated for a given model K by computing the maximum likelihood estimate (Dempster et al. 1977) ▸ Details...
- Note: the EM algorithm is sensitive to initialization, so we make use of a **splitting small-EM** initialization ▸ Details...

Classification by the MAP rule

"Maximum a posteriori" (MAP) rule:

- Each individual is attributed to the cluster for which it has the largest conditional probability of membership given the estimated parameters:

$$\tau_{ik}(\theta) = \frac{\pi_k f_k(\mathbf{y}_i|\theta_k)}{\sum_{\ell=1}^{K} \pi_\ell f_\ell(\mathbf{y}_i|\theta_\ell)}$$

- MAP rule with $\hat{\theta}_K$:

$$\hat{z}_{ik} = \begin{cases} 1 & \text{if } \tau_{ik}\left(\hat{\theta}_K\right) > \tau_{i\ell}\left(\hat{\theta}_K\right) \forall \ell \neq k \\ 0 & \text{otherwise} \end{cases}$$

Model selection

1. Collection of models $(\mathcal{S}_K)_{K\in\mathcal{K}}$ indexed by number of clusters K
2. In each model $\mathcal{S}_K$, parameter estimation via MLE: $\hat{\Psi}_K$
3. Selection of the "best" model $\hat{K}$ using a penalized criterion:

$$\hat{K} = \arg\min_{K\in\mathcal{K}} \left\{ -\frac{1}{n}\sum_{i=1}^{n} \log f(\mathbf{y}_i|K, \hat{\Psi}_K) + \text{penalty}(K) \right\}$$

$\Rightarrow$ Asymptotic penalized criteria include Bayesian Information Criterion (BIC) and Integrated Completed Likelihood (ICL) ▸ Details...

Slope heuristics for model selection (Birgé and Massart, 2006)

- Non-asymptotic framework: construct a penalized criterion[1] such that the selected model has a risk close to the oracle model
- Optimal penalty for model of dimension D:

$$\text{penalty}_{\text{opt}} \approx 2\kappa \frac{D}{n}$$

[1] Theoretically validated in Gaussian framework, but encouraging applications in other contexts (Baudry et al., 2012)

Slope heuristics for model selection (Birgé and Massart, 2006)

- Non-asymptotic framework: construct a penalized criterion[1] such that the selected model has a risk close to the oracle model
- Optimal penalty for model of dimension D:

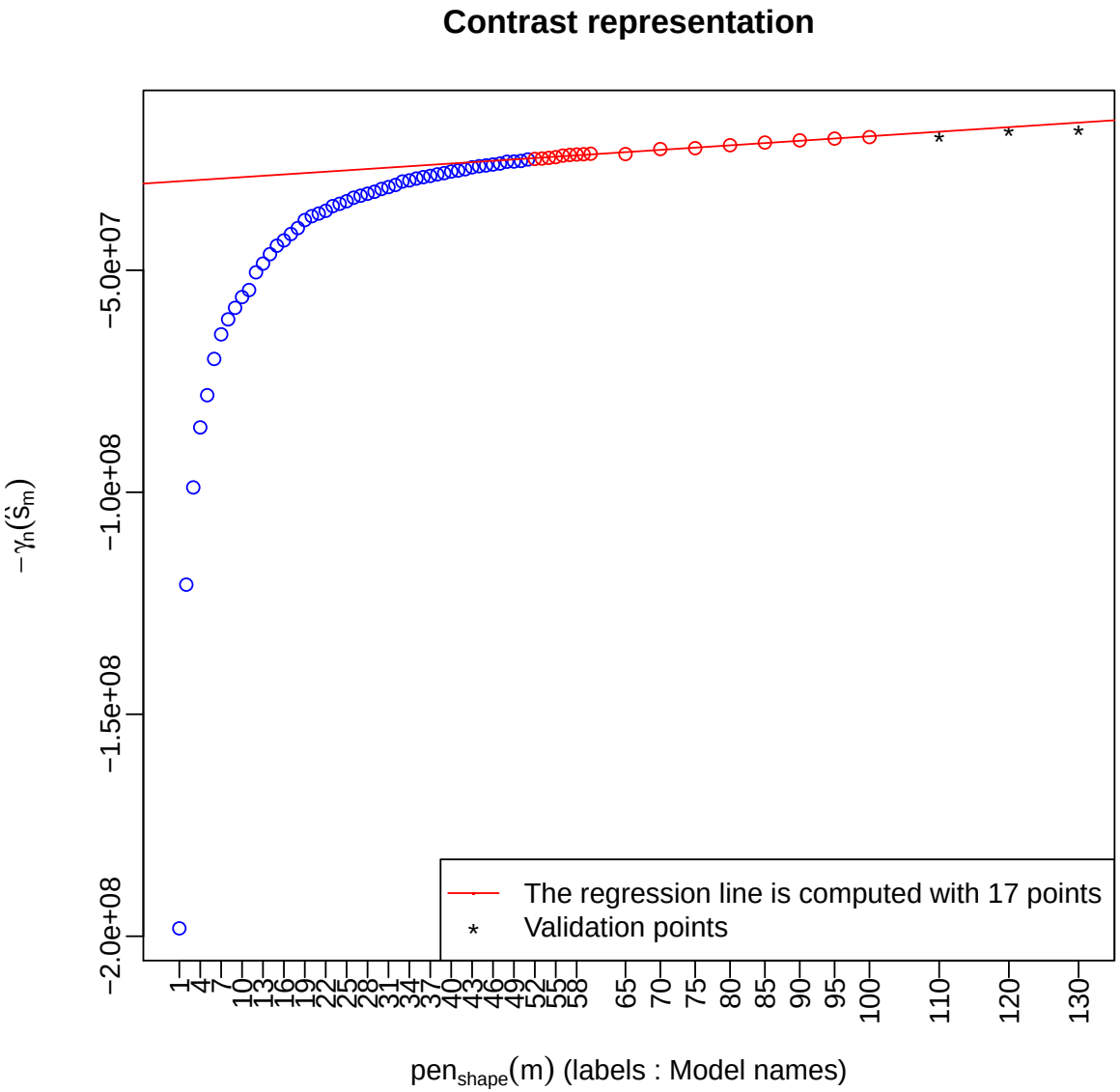
$$\text{penalty}_{\text{opt}} \approx 2\kappa \frac{D}{n}$$

In large dimensions:

- Linear behavior of loglikelihood with respect to model dimension D
- $\Rightarrow$ Estimation of slope to calibrate $\hat{\kappa}$ in a data-driven manner (Data-Driven Slope Estimation = DDSE), `capushe` R package

[1] Theoretically validated in Gaussian framework, but encouraging applications in other contexts (Baudry et al., 2012)

Slope heuristics in practice for RNA-seq



HTSCluster R package

```
> PMM <- PoisMixClusWrapper(y=data, gmin=1, gmax=35,
  conds=conds, split.init=TRUE, norm="TMM")
>
> summary(PMM)
**********************************************
Selected number of clusters via ICL = 10
Selected number of clusters via BIC = 30
Selected number of clusters via Djump = 15
Selected number of clusters via DDSE = 14
**********************************************
>
> summary(PMM$DDSE.results)
**********************************************
Number of clusters = 14
Model selection via DDSE
**********************************************
Cluster sizes:
Cluster 1 Cluster 2 Cluster 3 ...
540       192       235       ...
```

Real data analysis: Embryonic fly development

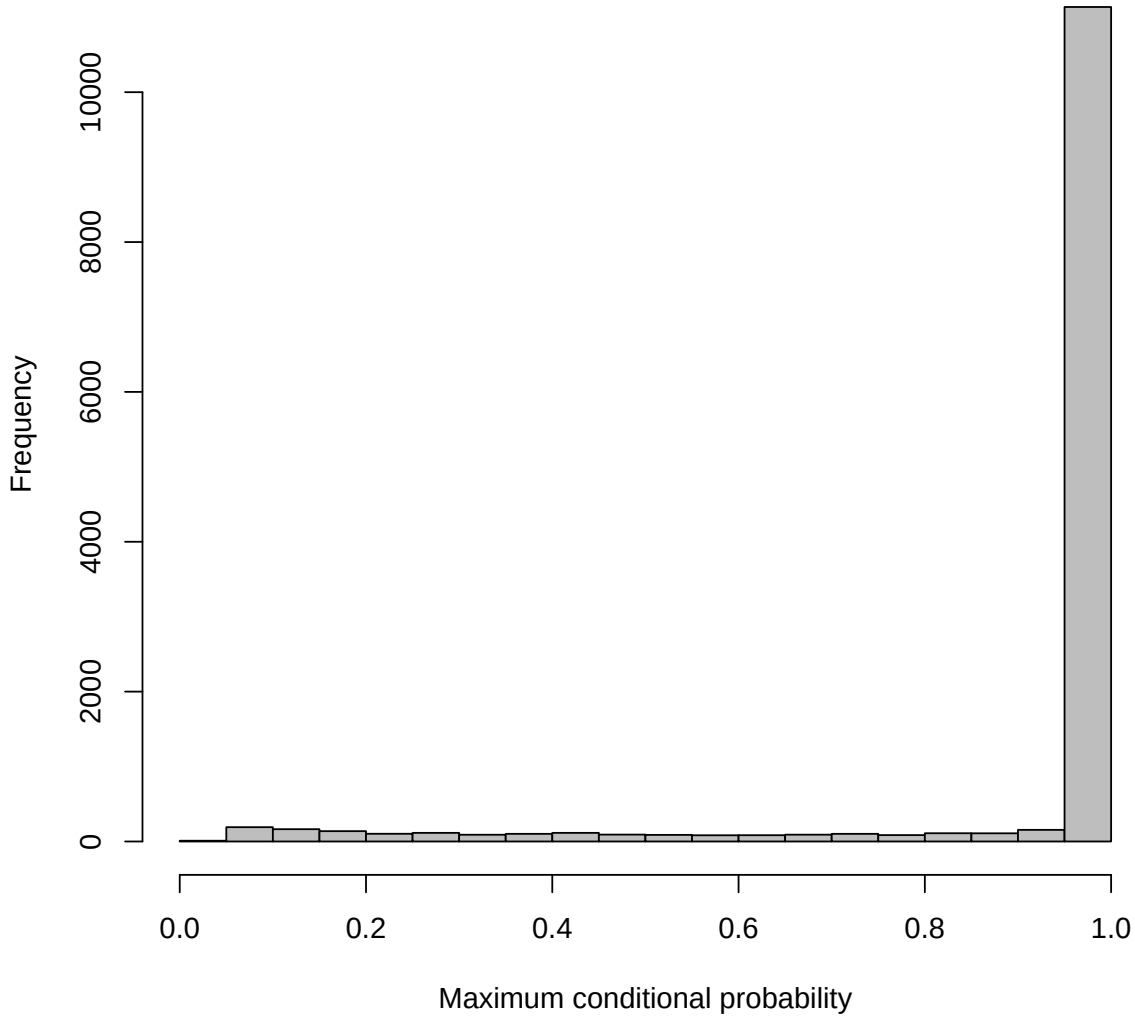
- modENCODE project to provide functional annotation of Drosophila (Graveley et al., 2011)
- Expression dynamics over 27 distinct stages of development during life cycle studied with RNA-seq
- 12 embryonic samples (collected at 2-hr intervals over 24 hrs) for 13,164 genes downloaded from ReCount database (Frazee et al., 2011)

Real data analysis: Embryonic fly development

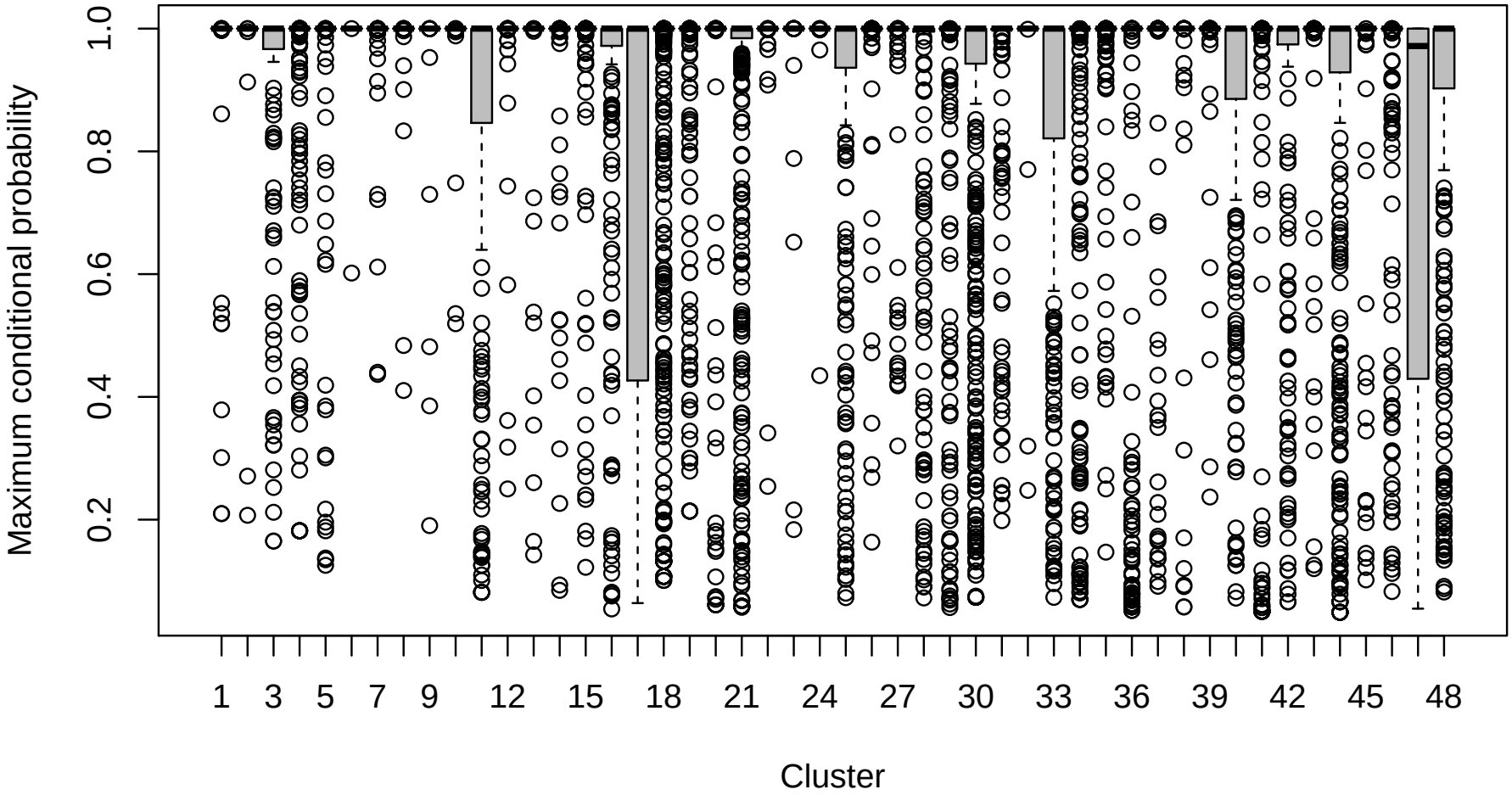
- modENCODE project to provide functional annotation of Drosophila (Graveley et al., 2011)
- Expression dynamics over 27 distinct stages of development during life cycle studied with RNA-seq
- 12 embryonic samples (collected at 2-hr intervals over 24 hrs) for 13,164 genes downloaded from ReCount database (Frazee et al., 2011)

- 3 independent runs, used `HTSCluster` to fit Poisson mixture models for $K \in \{1, \ldots, 60, 65, \ldots, 100, 110, \ldots, 130\}$
- Using slope heuristics, selected model is $\hat{K} = 48$

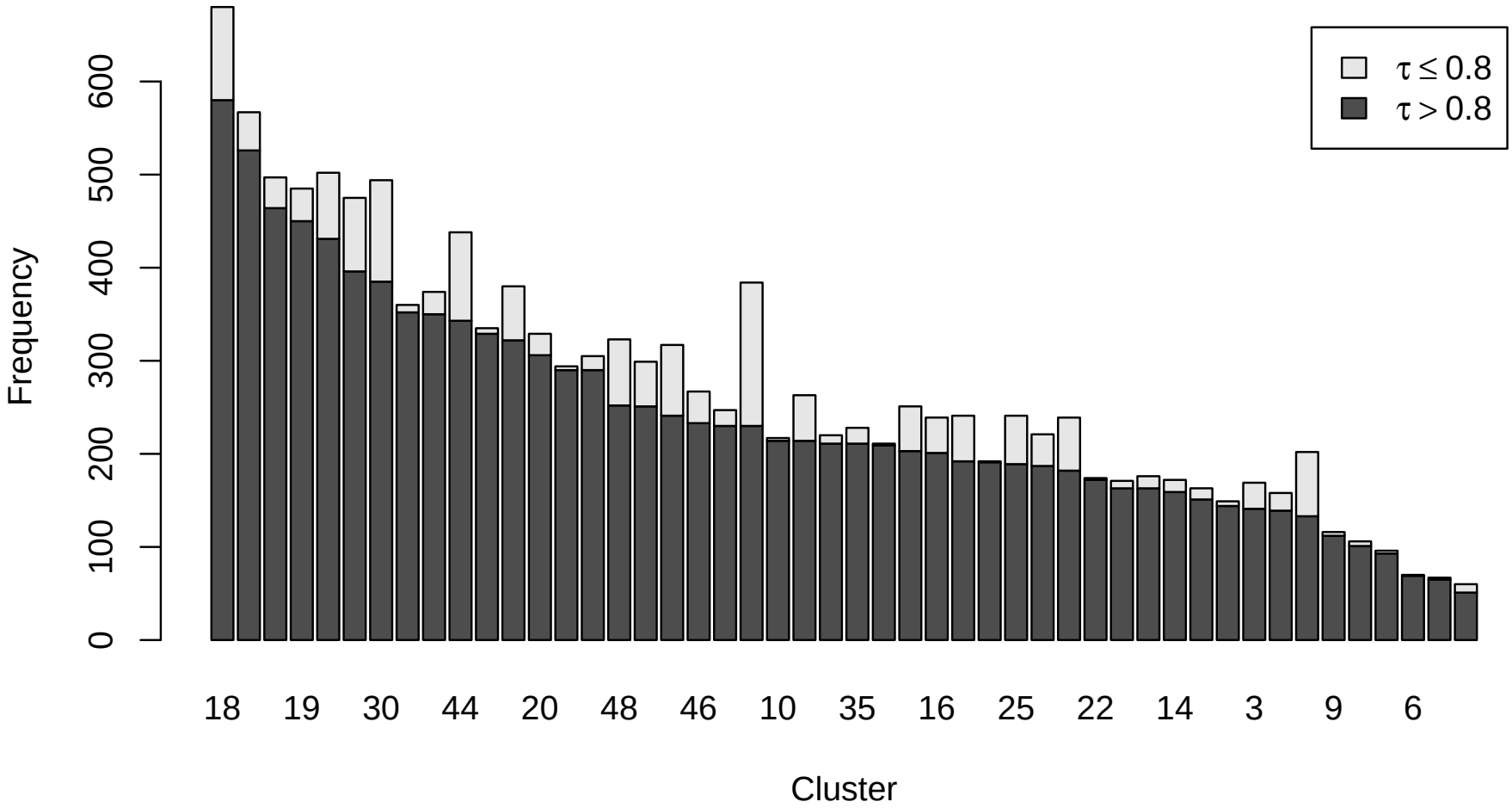
HTSCluster model diagnostics



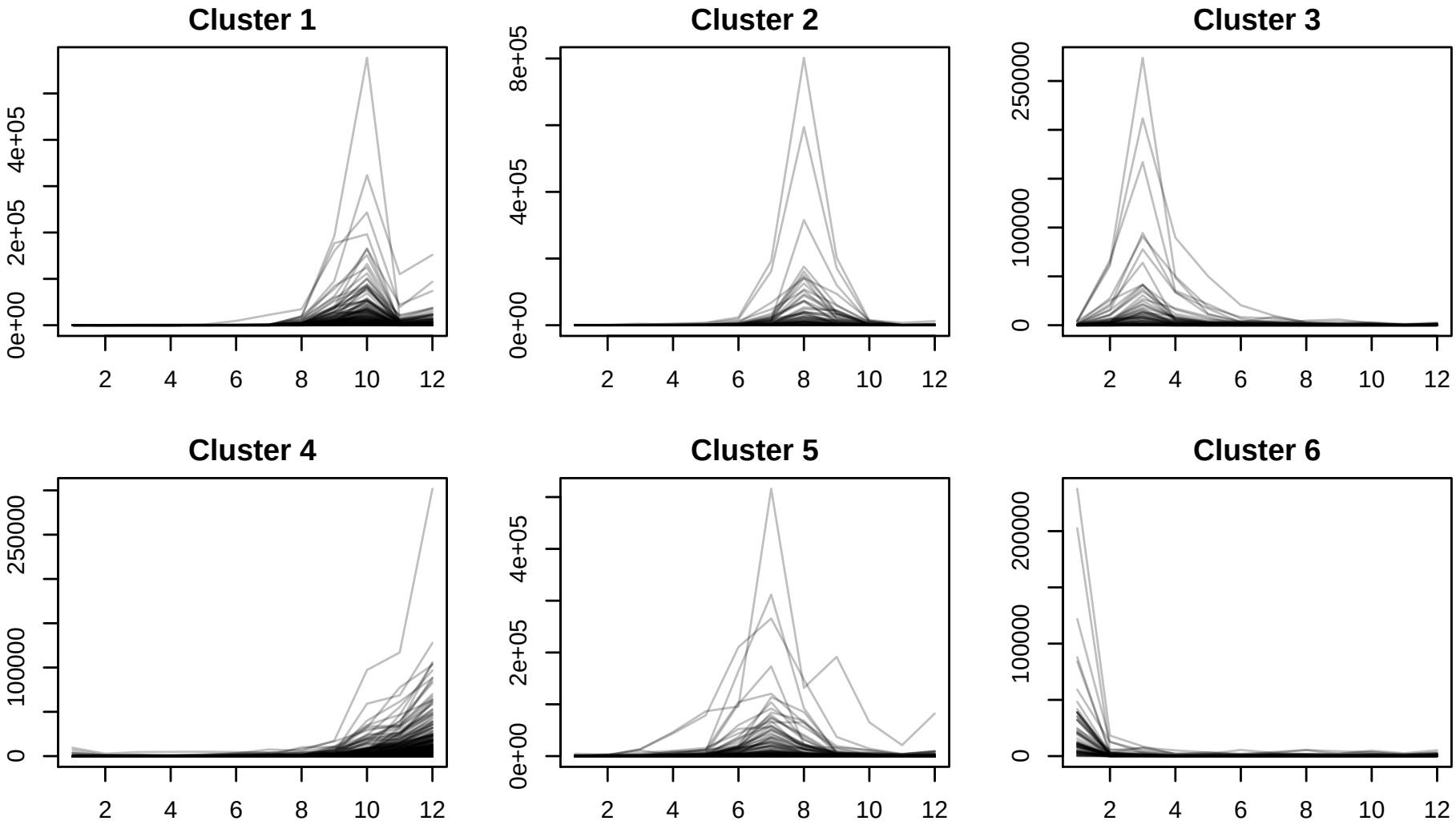
HTSCluster model diagnostics



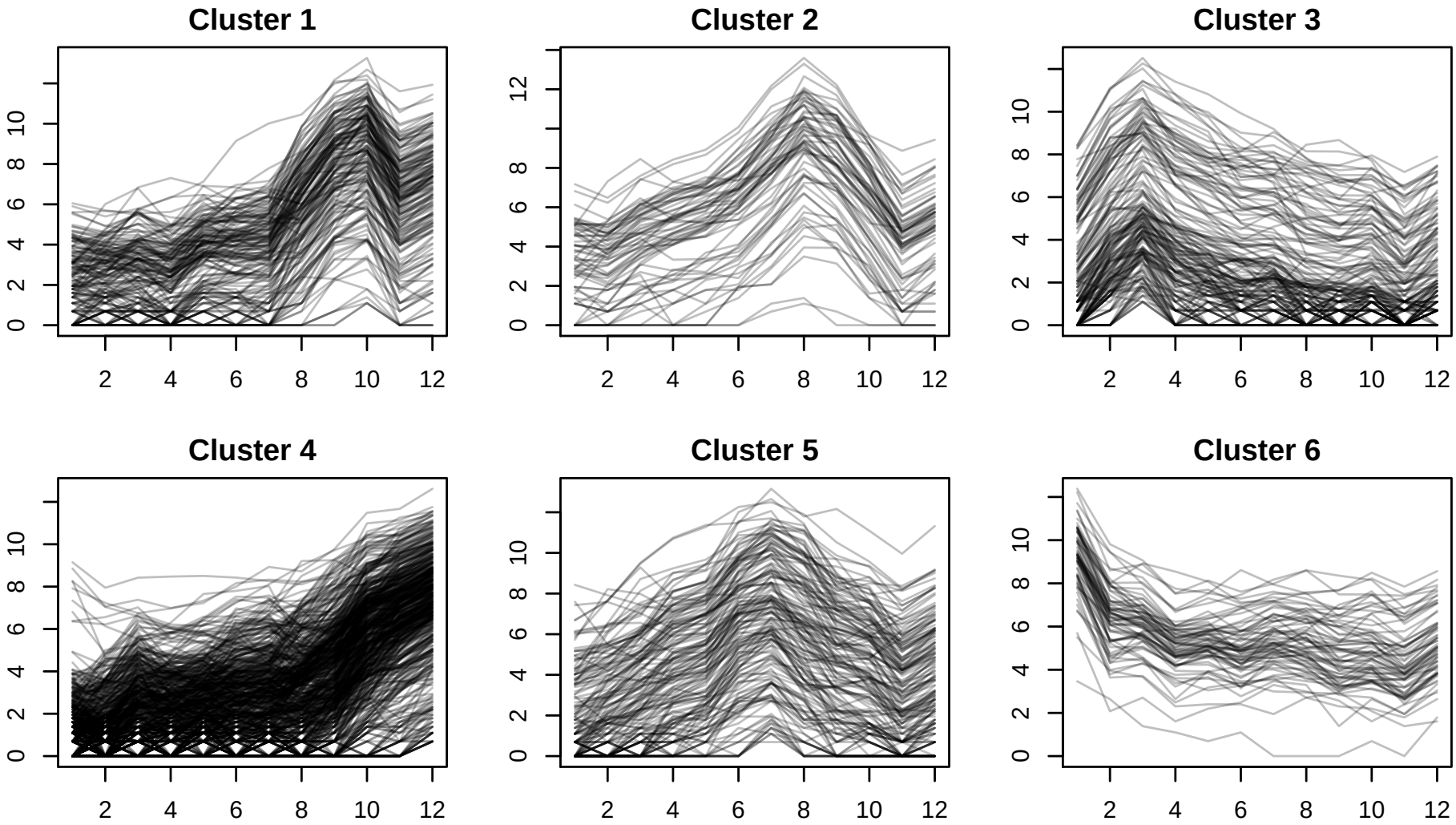
HTSCluster model diagnostics



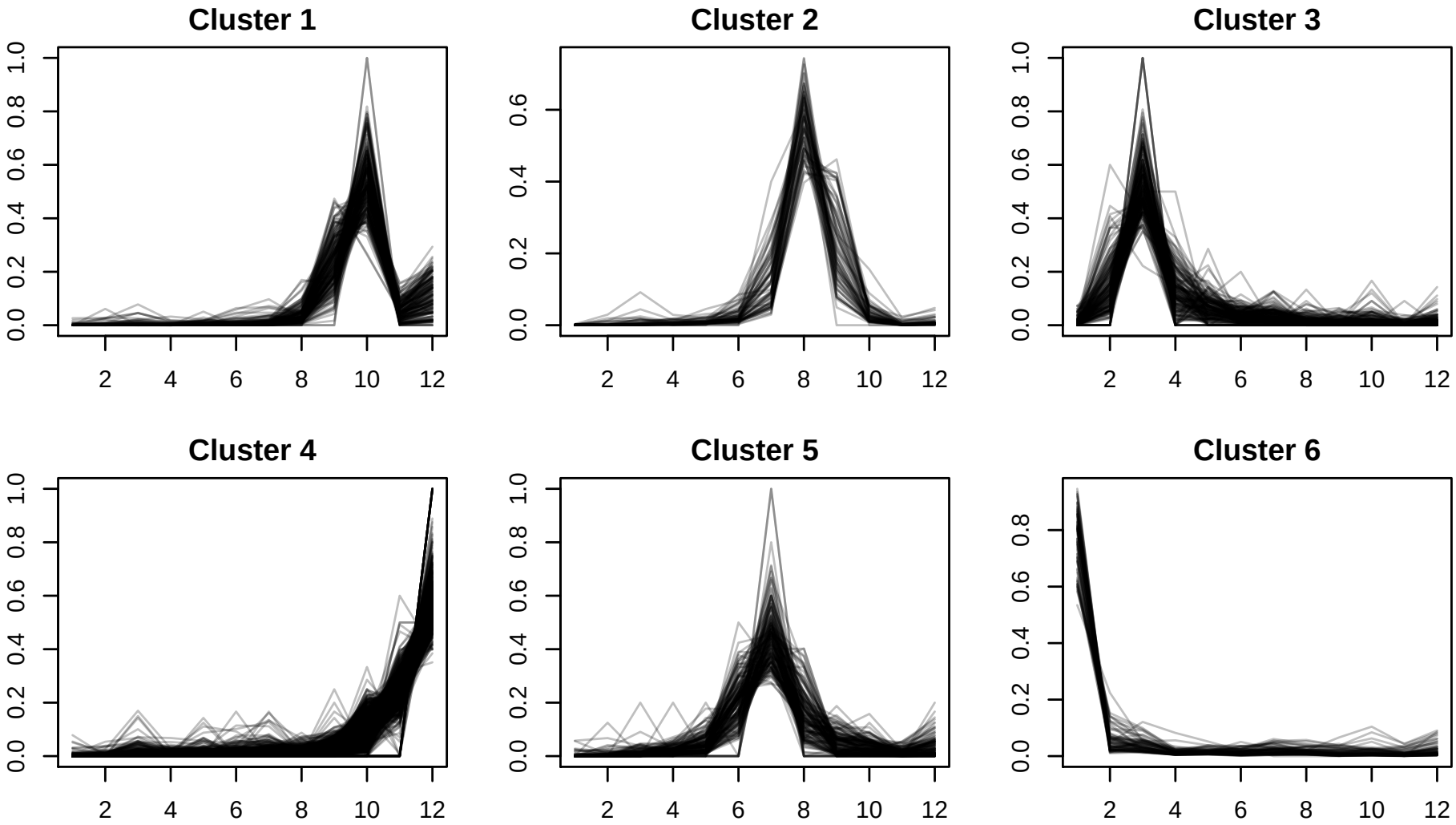
HTSCluster: Visualization of results



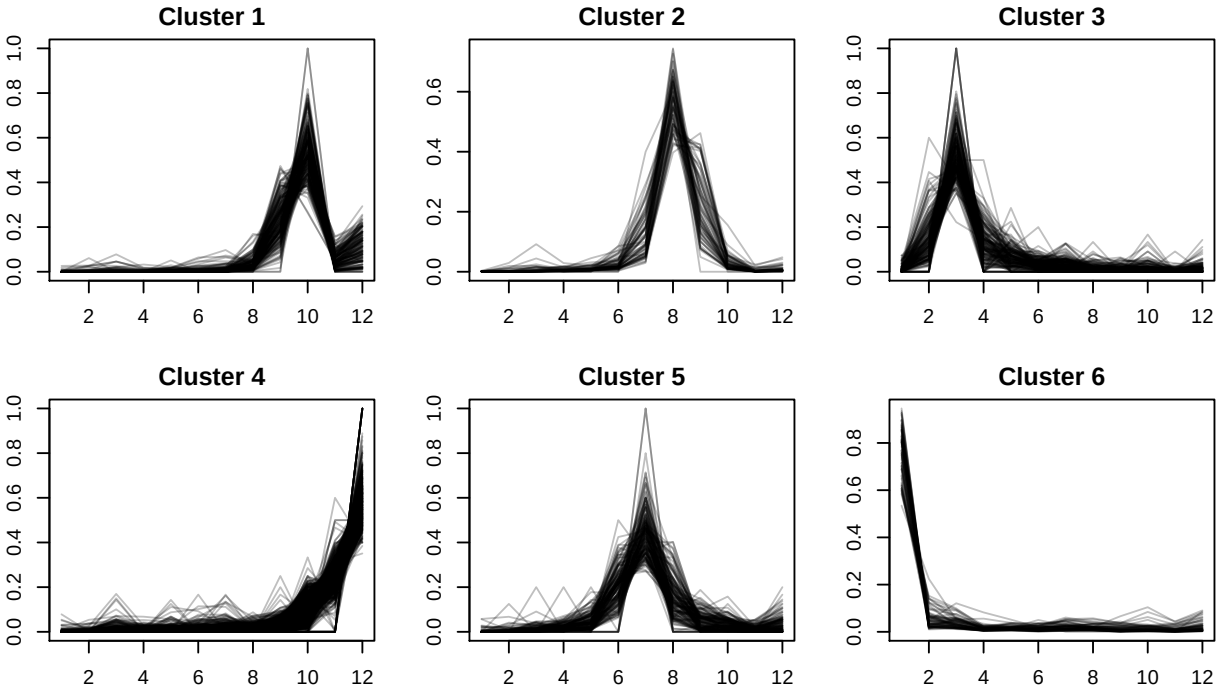
HTSCluster: Visualization of results

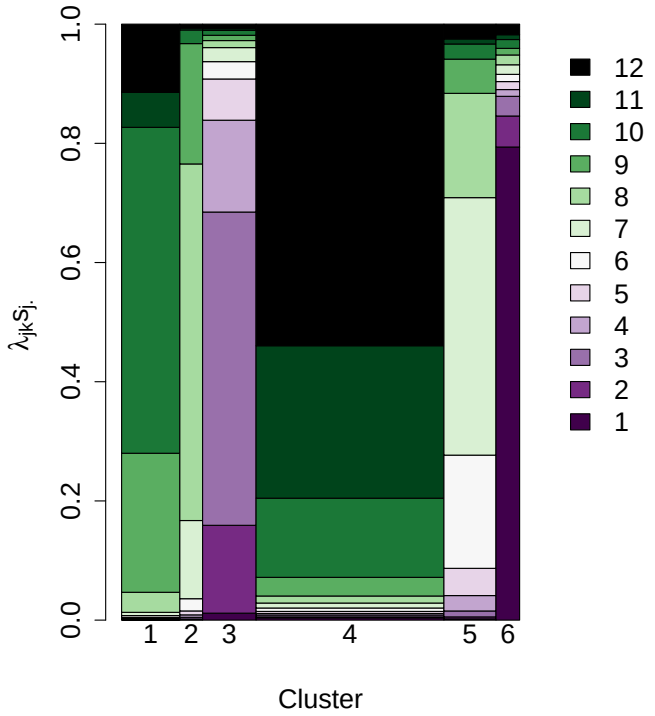


HTSCluster: Visualization of results

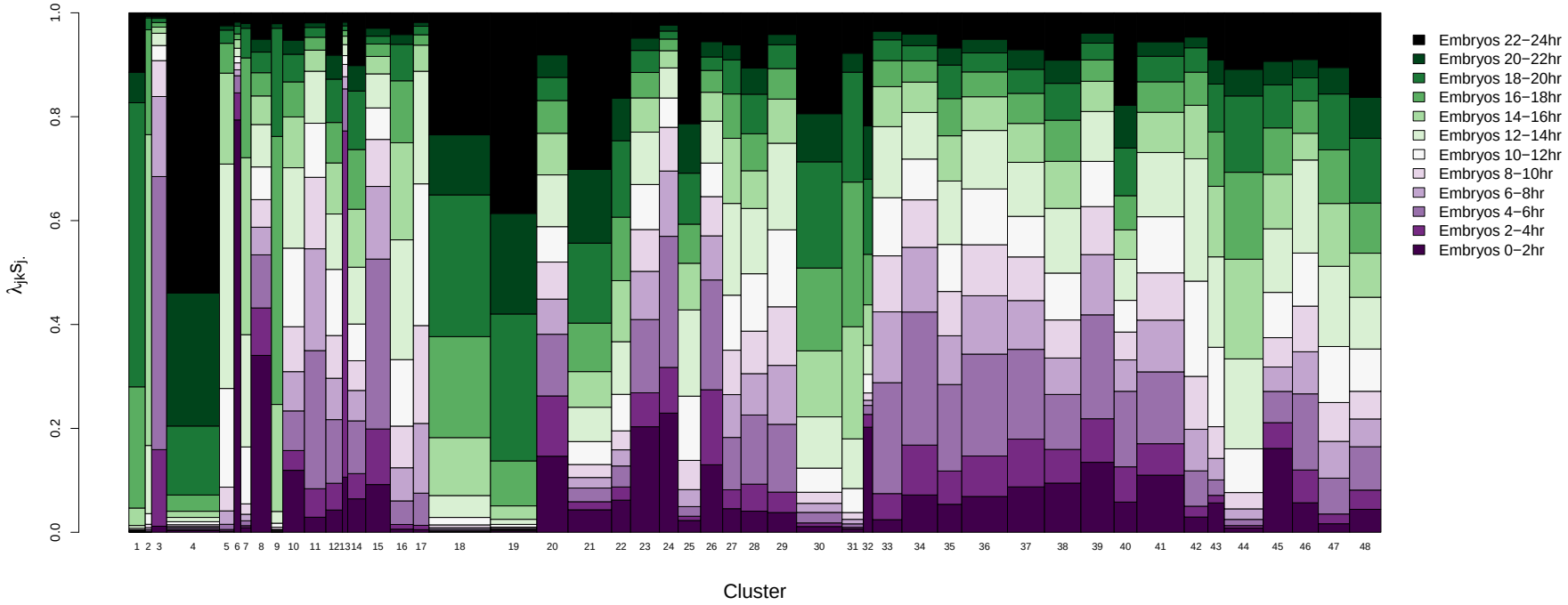


HTSCluster: Visualization of results





HTSCluster: Visualization of results



- Functional enrichment analysis: 33 of 48 clusters associated with at least one Gene Ontology Biological Process term (e.g., cluster 6 associated with muscle attachment)

HTSCluster for clustering count-based RNA-seq profiles

- Interpretable parameterization for RNA-seq co-expression analyses, straightforward parameter estimation, and a sound mechanism for model selection
- Performs well on real and simulated data compared to other approaches especially when the number of clusters is unknown
- HTSCluster (v2.0.4): R package on CRAN

BIOINFORMATICS

Co-expression analysis of high-throughput transcriptome sequencing data with Poisson mixture models

Andrea Rau [1,2]*, Cathy Maugis-Rabusseau [3], Marie-Laure Martin-Magniette[4,5,6,7] and Gilles Celeux [8]

Some limits (opportunities!) for `HTSCluster`

- Computational time can be a drawback: a full collection of models is estimated to allow for selection of a single "best" model, splitting small-EM initialization prevents parallelization...
- Samples are currently assumed to be conditionally independent given the cluster
- Conditions are currently assumed to be a single multi-level factor: how to correctly account for more complex experimental designs? (e.g. factorial, time series)
- Is a Poisson mixture model the most appropriate choice for RNA-seq data in practice? ...

Future work: Model comparisons for co-expression

Is it better to model the raw counts y_{ij} using a Poisson distribution or appropriately transformed counts $t(y_{ij})$ using a Gaussian distribution?[2]

$$f(\mathbf{y}_i|K,\theta_K) = \sum_{k=1}^{K} \pi_k \prod_{j=1}^{J} \mathcal{P}(y_{ij}|\boldsymbol{\theta}_k)$$

- vs -

$$g(t(\mathbf{y}_i)|K,\eta_K) = \sum_{k=1}^{K} \pi_k \Phi(t(\mathbf{y}_i)|\boldsymbol{\mu}_k,\Sigma_k)$$

[2] Ph.D. work of Mélina Gallopin

Future work: Model comparisons for co-expression

Is it better to model the raw counts y_{ij} using a Poisson distribution or appropriately transformed counts $t(y_{ij})$ using a Gaussian distribution?[2]

$$f(\mathbf{y}_i|K,\theta_K) = \sum_{k=1}^{K} \pi_k \prod_{j=1}^{J} \mathcal{P}(y_{ij}|\boldsymbol{\theta}_k)$$

- vs -

$$g(t(\mathbf{y}_i)|K,\eta_K) = \sum_{k=1}^{K} \pi_k \Phi(t(\mathbf{y}_i)|\boldsymbol{\mu}_k, \Sigma_k)$$

For example,

$$t(y_{ij}) = \log\left(\frac{y_{ij}/y_{\cdot j}+1}{m_i+1}\right)$$

[2] Ph.D. work of Mélina Gallopin

Future work: Model comparisons for RNA-seq co-expression

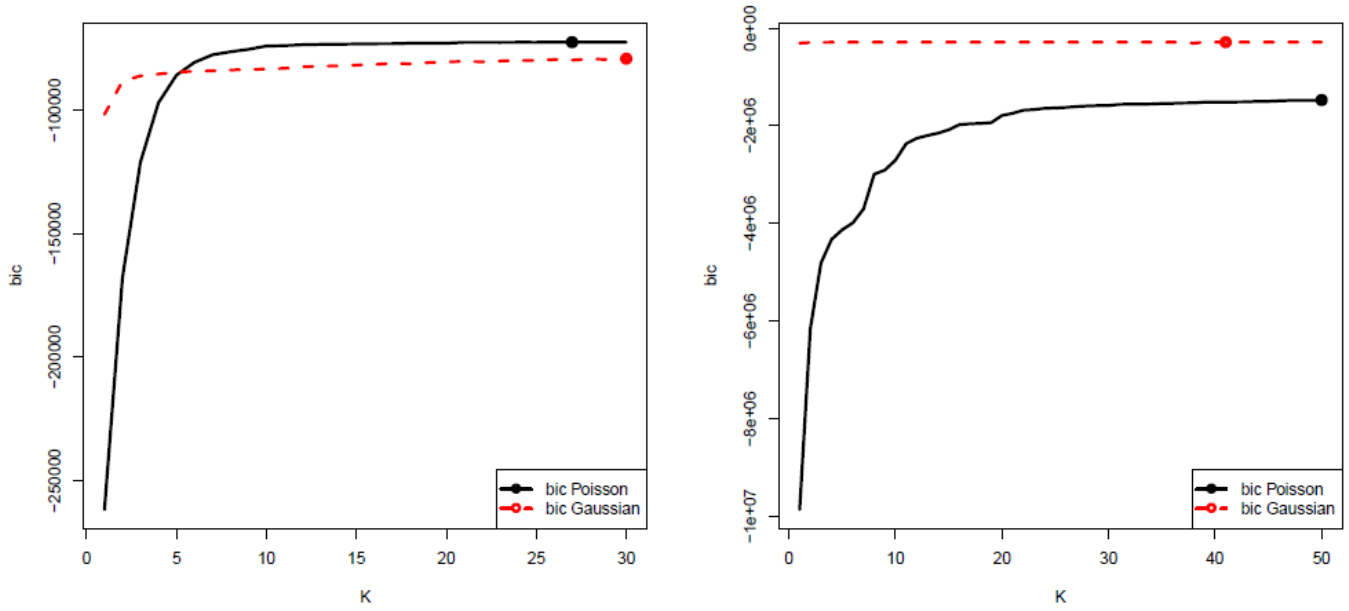
BIC model selection criterion enables an objective comparison:

- $\text{BIC}_f(K;\mathbf{y}) = \sum_{i=1}^{n} \log f(\mathbf{y}_i; K, \hat{\theta}_K) - \frac{\upsilon_f}{2} \log n$
- $\text{BIC}_g(K;\mathbf{y}) = \sum_{i=1}^{n} \log g(t(\mathbf{y}_i); K, \hat{\eta}_K) + \sum_{i=1}^{n} \log t'(\mathbf{y}_i) - \frac{\upsilon_g}{2} \log n$

Future work: Model comparisons for RNA-seq co-expression

BIC model selection criterion enables an objective comparison:

- $\text{BIC}_f(K;\mathbf{y}) = \sum_{i=1}^{n} \log f(\mathbf{y}_i; K, \hat{\theta}_K) - \frac{\upsilon_f}{2} \log n$
- $\text{BIC}_g(K;\mathbf{y}) = \sum_{i=1}^{n} \log g(t(\mathbf{y}_i); K, \hat{\eta}_K) + \sum_{i=1}^{n} \log t'(\mathbf{y}_i) - \frac{\upsilon_g}{2} \log n$



Left: Sultan *et al.* (2008). Right: Mach *et al.* (2014)

Further comparisons of transformations / models in progress...

Thank you!

In collaboration with...

- Gilles Celeux (Inria Saclay - Île-de-France)
- Cathy Maugis-Rabusseau (INSA / IMT Toulouse)
- Marie-Laure Martin-Magniette (AgroParisTech / INRA URGV)
- Panos Papastamoulis (University of Manchester)
- Mélina Gallopin (current Ph.D. student)

Estimation of finite mixture models

- A finite mixture model may be seen as an **incomplete data structure** model
- The **complete data** are

$$\mathbf{x} = (\mathbf{y}, \mathbf{z}) = (\mathbf{x}_1, \ldots, \mathbf{x}_n) = ((\mathbf{y}_1, \ldots, \mathbf{y}_n), (\mathbf{z}_1, \ldots, \mathbf{z}_n))$$

where the **missing data** are $\mathbf{z} = (\mathbf{z}_1, \ldots, \mathbf{z}_n) = (z_{ik})$

  - $\mathbf{z}_i$ component of i, where $z_{ik} = 1$ if i arises from group k and 0 otherwise
  - $\mathbf{z}$ defines a partition $P = (P_1, \ldots, P_K)$ of the observed data $\mathbf{y}$ with $P_k = \{i | z_{ik} = 1\}$

- Expected completed likelihood:

$$\mathcal{L}(\boldsymbol{\Psi}_K; \mathbf{y}, \mathbf{z}) = \sum_{i=1}^{n} \sum_{k=1}^{K} z_{ik} \left\{ \log \pi_k + \log f_k(\mathbf{y}; \boldsymbol{\theta}) \right\} + \lambda^{\pi} \left(\sum_{k=1}^{K} \pi_k - 1 \right)$$

where λ^{π} is the Lagrange multiplier for the constraint on $\boldsymbol{\pi}$

Estimation: EM algorithm (Dempster et al., 1977)

E-step Compute the conditional probabilities:

$$\tau_{ik}\left(\boldsymbol{\theta}_k^{(b)}\right) = \frac{\pi_k^{(b)} f(\mathbf{y}_i|\boldsymbol{\theta}_k^{(b)})}{\sum_{m=1}^{K} \pi_m^{(b)} f(\mathbf{y}_i|\boldsymbol{\theta}_m^{(b)})}$$

M-step Update $\boldsymbol{\Psi}_k$ to maximize the expected value of the completed likelihood by weighting observation i for cluster k with $\tau_{ik}\left(\boldsymbol{\theta}_k^{(b)}\right)$:

$$\hat{\pi}_k^{(b+1)} = \frac{1}{n}\sum_{i=1}^{n} \tau_{ik}\left(\boldsymbol{\theta}_k^{(b)}\right),$$

$$\hat{w}_i = y_{i\cdot\cdot}$$

$$\hat{\lambda}_{jk}^{(b+1)} = \frac{\sum_{i=1}^{n} \tau_{ik}\left(\boldsymbol{\theta}_k^{(b)}\right) y_{ij\cdot}}{\hat{s}_{j\cdot} \sum_{i=1}^{n} \tau_{ik}\left(\boldsymbol{\theta}_k^{(b)}\right) y_{i\cdot\cdot}}$$

◂ Back...

Splitting initialization (Papastamoulis et al., 2014)

for $K \leftarrow 2$ **to** $gmax$ **do**

- Calculate per-class entropy $e_k = -\sum_{i \in k} \log \hat{t}_{ik}^{K-1}$ for model with $(K-1)$ clusters
- Select cluster $k^\star = \arg\max_k e_k$ to be split
- **for** $i \leftarrow 1$ **to** $init.runs$ **do**
  - Randomly split the observations in cluster $k^\star$ into two clusters
  - Calculate corresponding $\lambda^{(0,i),K}$ and $\pi^{(0,i),K}$
  - Update values of $\lambda^{(0,i),K}$ and $\pi^{(0,i),K}$ via EM algorithm with `init.iter` iterations
  - Calculate the log-likelihood $L^{(i),K} = L(\hat{\lambda}^{(0,i),K}, \hat{\pi}^{(0,i),K})$
- **end**
- Let $i^\star = \arg\max_i L^{(i),K}$. Fix new initial values $\lambda^{(0),K} = \hat{\lambda}^{(0,i^\star),K}$ and $\pi^{(0),K} = \hat{\pi}^{(0,i^\star),K}$.

end

Back...

Penalized criteria for model selection: BIC (Schwarz, 1978)

- Maximization of integrated likelihood:

$$\hat{m} = \arg\min_{m \in \mathcal{M}} -f(\mathbf{y}|m)$$

where

$$f(\mathbf{y}|m) = \int_{\Theta_m} f(\mathbf{y}|\theta, m)\Pi(\theta|m)d\theta$$

- Asymptotic approximation (where $D_m = (m-1) + m \times J$ is the dimension of $\mathcal{S}_m$):

$$\begin{aligned} -\ln(f(\mathbf{y}|m)) &\approx -L(\mathbf{y}|\hat{\theta}_m) + \frac{D_m}{2}\ln(n) \\ &= n\gamma_n(\hat{s}_m) + \frac{D_m}{2}\ln(n) \end{aligned}$$

- Bayesian information criterion (BIC):

$$\hat{m} = \arg\min_{m \in \mathcal{M}} \{\gamma_n(\hat{s}_m) + \text{pen}_{\text{BIC}}(m)\} \text{ with } \text{pen}_{\text{BIC}}(m) = \frac{D_m}{2n}\ln(n)$$

Back...

Penalized criteria for model selection: ICL (Biernacki et al., 2000)

- Alternative based on maximization of integrated completed likelihood:

$$\hat{m} = \arg\min_{m \in \mathcal{M}} -f(\mathbf{y}, \mathbf{z}|m)$$

where

$$f(\mathbf{y}, \mathbf{z}|m) = \int_{\Theta_m} f(\mathbf{y}, \mathbf{z}|\theta, m)\Pi(\theta|m)d\theta$$

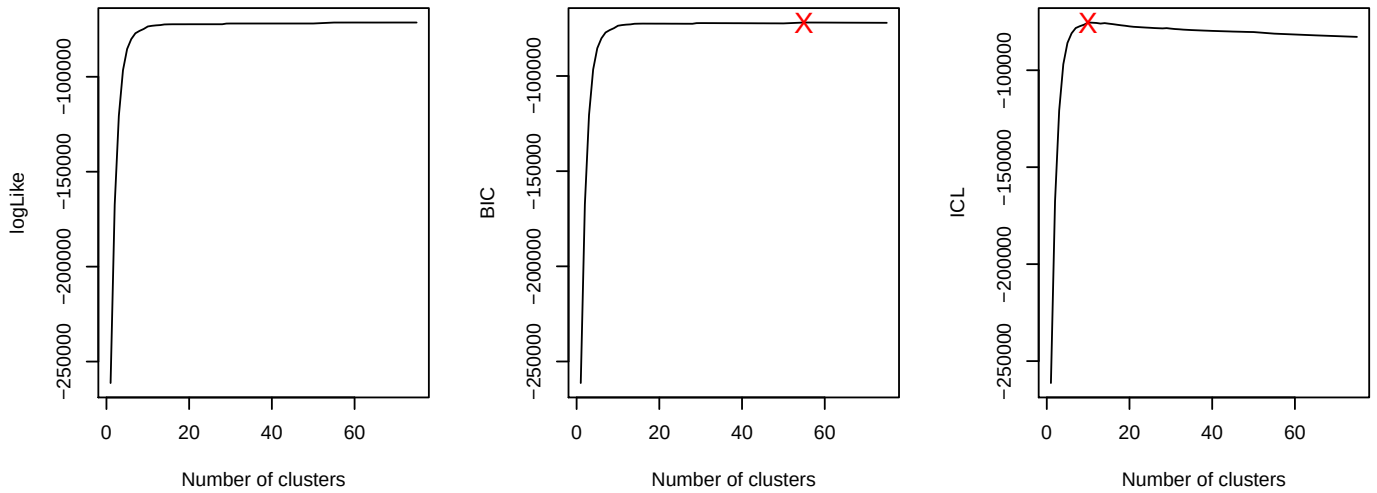
- BIC-like asymptotic approximation for Integrated Completed Likelihood (ICL):

$$\begin{aligned}\hat{m} &= \arg\min_{m \in \mathcal{M}} \{-\frac{1}{n} L(\hat{\theta}_m|\mathbf{y}, \hat{\mathbf{z}}) + \frac{D_m}{2n}\ln(n)\} \\ &= \arg\min_{m \in \mathcal{M}} \{\gamma_n(\hat{s}_m) + \text{pen}_{\text{BIC}}(m) + \text{Ent}(m)\}\end{aligned}$$

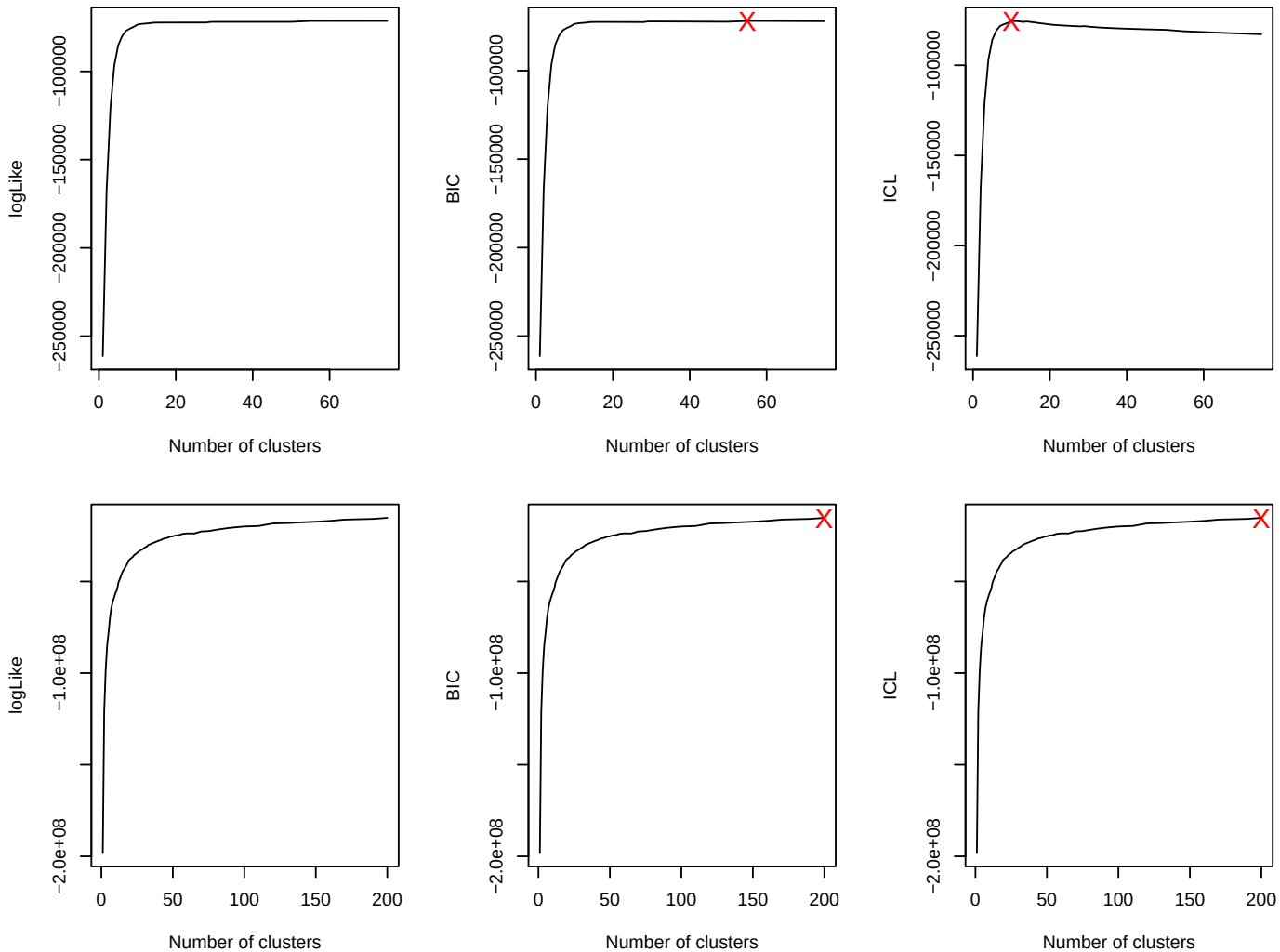
where $\text{Ent}(m) = -\frac{1}{n}\sum_i \sum_k \hat{z}_{ik} \ln \tau_{ik}(\hat{\theta}_m)$

Back...

Behavior of BIC and ICL in practice for RNA-seq data



Behavior of BIC and ICL in practice for RNA-seq data



Back...

Description of competing models

1. PoisL (Cai et al., 2004): K-means type algorithm using Poisson loglinear model
   - Equivalent to `HTSCluster` when equal library sizes, unreplicated data, equiprobable Poisson mixtures, and parameter estimation via the Classification EM (CEM) algorithm
2. Witten (2011): hierarchical clustering of dissimilarity measure based on a Poisosn loglinear model
   - Originally intended to cluster samples
3. Si et al. (2014): model-based hierarchical algorithm using Poisson and negative binomial models
4. Classic K-means algorithm on expression profiles ($y_{ij\ell}/y_{i\cdot\cdot}$)

Model selection not addressed by any of the above $\Rightarrow$ Caliński and Harabasz index (1974) used for comparison

Simulation procedure (based on fly and human data)

For each setting (fly and human), 50 individual datasets:

- K fixed to 15, true experimental design used
- All parameters (λ_{jk}), $(s_{j\ell})$, and (w_i) fixed to estimated values from real data analysis
- $n = 3000$ genes randomly sampled from fly or human data, weighted by their maximum conditional probability
- For each selected gene, we sample from the appropriate Poisson distribution:

$$Y_{ijl} \sim \mathcal{P}(\mu_{ijlk})$$

where $\mu_{ijlk} = w_i s_{jl} \lambda_{jk}$ if $\hat{z}_{ik} = 1$.

Simulation results

- All models fit for $K \in 1, \ldots, 40$
- Model selection via the slope heuristics (`HTSCluster`, PoisL) or CH-index (Si-Pois, Si-NB, Witten)
- Models compared using the adjusted Rand Index (ARI, Hubert & Arabie 1985)
- For comparison, also consider the oracle ARI (based on assignment of observations to clusters using the true parameter values)

Simulation results

Table: Mean (sd) ARI for simulations with parameters based on the fly and human liver.

Method	Model selection	Fly	Human
`HTSCluster`	`capushe`	**0.93** (0.05)	**0.61** (0.02)
	True K	0.84 (0.09)	**0.60** (0.02)
PoisL	`capushe`	0.79 (0.15)	0.53 (0.05)
	True K	0.82 (0.05)	0.53 (0.04)
Witten	CH index	0.15 (0.07)	0.11 (0.03)
	True K	0.67 (0.09)	0.39 (0.04)
Si-Pois	CH index	0.26 (0.17)	0.48 (0.04)
	True K	**0.95** (0.02)	**0.61** (0.02)
Si-NB	CH index	0.23 (0.16)	0.47 (0.04)
	True K	**0.94** (0.02)	**0.60** (0.02)
K-means	True K	0.79 (0.08)	0.42 (0.02)
Oracle	True K	**0.95** (0.01)	**0.63** (0.01)

◂ Back...