Regularized Maximum-Likelihood Estimation of Mixture-of-Experts

Faicel Chamroukhi & Bao-Tuyen Huynh

https://chamroukhi.users.lmno.cnrs.fr

IJCNN 2018, Rio

July 11, 2018
Scientific context

- Heterogeneous regression data $\rightarrow$ underlying unknown partition
- Data issued from non-linear regression function

Modeling framework

- **Latent variable models**: $f(x|\theta) = \int_z f(x, z|\theta)dz$

  **Generative formulation**:

  $z \sim q(z|\theta)$

  $x|z \sim f(x|z, \theta)$
Outline

1. Mixture-of-Experts (MoE) Modeling and MLE
2. Regularized MLE of the MoE
3. Proposed EM algorithm with block coordinate ascent
4. Experimental study
Mixture-of-Experts (MoE) modeling framework

- Observed pairs of data \((x, y)\) where the response \(y \in \mathbb{R}\) for the predictors \(x \in \mathbb{R}^p\) governed by a hidden categorical random variable \(Z\)

- Mixture of experts (MoE) (Jacobs et al., 1991; Jordan and Jacobs, 1994):

  \[
  f(y|x; \theta) = \sum_{k=1}^{K} \pi_k(x; w) f_k(y|x; \theta_k)
  \]

  - Gating network (e.g softmax):
    \[
    \pi_k(x; w) = \frac{\exp (w_{k0} + w_k^T r)}{1 + \sum_{\ell=1}^{K-1} \exp (w_{\ell0} + w_{\ell}^T r)}
    \]

  - Experts network (e.g Gaussian regressors):
    \[
    f_k(y|x; \theta_k) = \phi(y; \mu(x; \beta_k), \sigma_k^2)
    \]
    with parametric (non-)linear regression functions \(\mu(x; \beta_k)\)

  - is parameterized by \(\theta = (w^T, \theta_1^T, \ldots, \theta_K^T)^T\)

- Non-normal MoE, for data with atypical observations, and with possible heavy tailed and asymmetric distributions: Chamroukhi (2016, 2017); Nguyen and Chamroukhi (2018)
Standard MLE of the MoE model

MLE: $\theta$ is commonly estimated by maximizing the observed-data log-likelihood:

$$\hat{\theta}_n \in \arg \max_{\theta \in \Theta} L(\theta)$$

with

$$L(\theta) = \ln f((x_1, y_1), \ldots, (x_n, y_1); \theta) = \sum_{i=1}^n \ln \sum_{k=1}^K \pi_k(x_i; w) f(y_i | x_i; \theta_k).$$

$\hookrightarrow$ the EM algorithm (Dempster et al. (1977))
Standard MLE of the MoE model

- MLE: $\theta$ is commonly estimated by maximizing the observed-data log-likelihood:
  $$\hat{\theta}_n \in \arg \max_{\theta \in \Theta} L(\theta)$$

  with
  $$L(\theta) = \ln f((x_1, y_1), \ldots, (x_n, y_1); \theta) = \sum_{i=1}^n \ln \sum_{k=1}^K \pi_k(x_i; w) f(y_i | x_i; \theta_k).$$

  $\hookrightarrow$ the EM algorithm (Dempster et al. (1977))

  $\hookrightarrow$ The standard MLE of MoE when $p$ is large (high-dimensional setting)
  $\hookrightarrow$ the features are possibly correlated and sparse
  $\hookrightarrow$ Looking for a sparse models

Regularized MLE of the MoE

RMLE: $\theta$ is estimated by maximizing a penalized observed-data log-likelihood:
  $$\hat{\theta}_n \in \arg \max_{\theta \in \Theta} PL(\theta)$$

  with $PL(\theta) = L(\theta) - \text{Pen}(\theta)$

  $\hookrightarrow$ $\text{Pen}(\theta)$ should encourage sparsity
  $\hookrightarrow$ parameter estimation and selection problem
Proposed Regularized Mixture of Experts model

\[
\text{Pen}(\theta) = \sum_{k=1}^{K} \lambda_k \|\beta_k\|_1 + \sum_{k=1}^{K-1} \gamma_k \|\mathbf{w}_k\|_1 + \frac{\rho}{2} \|\mathbf{w}_k\|_2^2
\]

- Lasso penalty for the experts \(\rightarrow\) encourage a sparse solution
- The elastic net penalty (Zou and Hastie (2005)) for the gating network:
  \(\rightarrow\) reduce the norm of the estimated values of the gating network parameters by using the \(L_2\) penalties;
  \(\rightarrow\) the Lasso penalty to recover a sparse solution
- The convexity of \(L_1\) and \(L_2\) penalties have also advantageous numerical properties.
- If the correlation between the features is high, one can add \(L_2\) penalties for the expert network.
The penalized log-likelihood function:

\[ PL(\theta) = L(\theta) - \sum_{k=1}^{K} \lambda_k \| \beta_k \|_1 - \sum_{k=1}^{K-1} (\gamma_k \| w_k \|_1 + \frac{\rho}{2} \| w_k \|_2^2) \] (1)

The penalized complete-data log-likelihood function:

\[ PL_c(\theta) = L_c(\theta) - \sum_{k=1}^{K} \lambda_k \| \beta_k \|_1 - \sum_{k=1}^{K-1} (\gamma_k \| w_k \|_1 + \frac{\rho}{2} \| w_k \|_2^2) \] (2)

with

\[ L_c(\theta) = \ln f((x_1, y_1, z_1), \ldots, (x_n, y_n, z_n); \theta) = \sum_{i=1}^{n} \sum_{k=1}^{K} z_{ik} \log [\pi_k(x_i; \theta) f(y_i| x_i; \theta_k)] \]

such that \( z_{ik} = 1 \) iff \( z_i = k \) (the data pair \( (x_i, y_i) \) originates from expert \( k \))
Statistical inference for RMoE

**Theorem (Khalili (2010))**

Let \((V_i)_{i=1,...,n} = (X_i, Y_i)_{i=1,...,n}\) be a random sample from a density function \(f(v; \theta)\) \((\theta = (\theta_1, \theta_2, \ldots, \theta_h))\) which satisfies some regularity conditions:

The joint density of \(V_i\) is given by

\[
    f(v_i; \theta) = f(x_i) \sum_{k=1}^{K} \pi_k(x_i; w)p(y_i|x_i; \theta_k).
\]

Assume that \(\rho/\sqrt{n} \to 0\) as \(n \to \infty\). Then, there exists a local maximizer \(\hat{\theta}_n\) of the regularized log-likelihood function \(PL(\theta)\) (1) for which

\[
    ||\hat{\theta}_n - \theta_0|| = O\left(\frac{1}{\sqrt{n}} (1 + q_1^* + q_1 n)\right),
\]

where

\[
    q_1^* = \max_{k,j}\{\lambda_k/\sqrt{n} : \beta_{k,j}^0 \neq 0\}; \quad q_1 n = \max_{k,j}\{\gamma_k/\sqrt{n} : \omega_{k,j}^0 \neq 0\}.
\]

- By choosing \(\max_k \gamma_k = O(\sqrt{n})\), \(\max_k \lambda_k = O(\sqrt{n})\) we have the root-\(n\) consistent property for \(\hat{\theta}_n\).
Parameter estimation for RMoE

Khalili’s method:

- Approximates the $L_1$ penalty function in a some neighborhood by an $\varepsilon$-local quadratic function
  \[ \eta|t| \approx \eta|t_0| + \frac{\eta}{2(|t_0| + \varepsilon)} (t^2 - t_0^2). \]

  ⇝ Almost surely none of the components will be exactly zero.

- Needs using a threshold to recover the zero coefficients
  ⇝ The size of threshold affects the degree of sparsity of the solution.

- The Newton-Raphson algorithm is used to update the M-step of the EM algorithm.
  ⇝ This approach still require computing the inverse matrix.

In our proposal:

- A block EM algorithm with coordinate ascent algorithm to estimate the parameters:
  ⇝ Exact $L_1$ penalty regularization;
  ⇝ Avoids computing matrix inversion;
  ⇝ Avoids using a threshold to recover the zero coefficients.
Block EM algorithm with coordinate ascent

E-step

- Compute the conditional expectation of the penalized complete-data log-likelihood

\[
Q(\theta; \theta^{(q)}) = \mathbb{E} \left[ P L_c(\theta) | D; \theta^{(q)} \right] 
= \sum_{i=1}^{n} \sum_{k=1}^{K} \tau_{ik}^{(q)} \log \left[ \pi_k(x_i; w_k) f_k(y_i|x_i; \theta_k) \right] 
- \sum_{k=1}^{K} \lambda_k \| \beta_k \|_1 
- \sum_{k=1}^{K-1} (\gamma_k \| w_k \|_1 - \frac{\rho}{2} \| w_k \|_2^2). 
\]
Block EM algorithm with coordinate ascent

E-step

- Compute the conditional expectation of the penalized complete-data log-likelihood

\[
Q(\theta; \theta^{(q)}) = \mathbb{E} \left[ P L_c(\theta) | \mathcal{D}; \theta^{(q)} \right]
\]

\[
= \sum_{i=1}^{n} \sum_{k=1}^{K} \tau_{ik}^{(q)} \log \left[ \pi_k (x_i; w) f_k (y_i | x_i; \theta_k) \right]
\]

\[
- \sum_{k=1}^{K} \lambda_k \| \beta_k \|_1 - \sum_{k=1}^{K-1} \left( \gamma_k \| w_k \|_1 - \frac{\rho}{2} \| w_k \|_2^2 \right).
\]

→ Calculate the posterior component probabilities:

\[
\tau_{ik}^{(q)} = \mathbb{P}(Z_i = k | y_i, x_i; \theta^{(q)}) = \frac{\pi_k (x_i; w^{(q)}) N(y_i; \beta_k^{(q)} + x_i \beta_k^{(q)}, \sigma_k^{(q)^2})}{\sum_{l=1}^{K} \pi_l (x_i; w^{(q)}) N(y_i; \beta_l^{(q)} + x_i \beta_l^{(q)}, \sigma_l^{(q)^2})}.
\]

→ As in standard MoE
**M-step**

- Maximizing the $Q$ function: $\theta^{(q+1)} \in \arg\max_{\theta} Q(\theta; \theta^{(q)})$ with

$$Q(\theta; \theta^{(q)}) = Q(w; \theta^{(q)}) + Q(\beta, \sigma; \theta^{(q)}),$$

where

$$Q(w; \theta^{(q)}) = \sum_{i=1}^{n} \sum_{k=1}^{K} \tau_{ik}^{(q)} \log \pi_k(x_i; w) - \sum_{k=1}^{K-1} (\gamma_k \| w_k \|_1 - \frac{\rho}{2} \| w_k \|_2^2), \quad (3)$$

$\rightarrow$ a weighted regularized multiclass logistic regression problem

and

$$Q(\beta, \sigma; \theta^{(q)}) = \sum_{k=1}^{K} \sum_{i=1}^{n} \tau_{ik}^{(q)} \log \mathcal{N}(y_i; \beta_{k0} + x_i^T \beta_k, \sigma_k^2) - \sum_{k=1}^{K} \lambda_k \| \beta_k \|_1 \quad (4)$$

$\rightarrow$ $K$ independent weighted LASSO problems
Updating the gating network parameters

- Coordinate ascent algorithm to update $w$ Tseng (1988, 2001)
- $w_{kj}$ is updated by maximizing the component $(k, j)$ of (3) given by

$$Q(w_{kj}; \theta^{(q)}) = \begin{cases} F(w_{kj}; \theta^{(q)}) - \gamma_k w_{kj}, & \text{if } w_{kj} > 0 \quad (F_1) \\ F(0; \theta^{(q)}) & \text{if } w_{kj} = 0, \\ F(w_{kj}; \theta^{(q)}) + \gamma_k w_{kj}, & \text{if } w_{kj} < 0 \quad (F_2) \end{cases}$$

$$F(w_{kj}; \theta^{(q)}) = \sum_{i=1}^{n} \tau_{ik}^{(q)}(w_k + w^T_k x_i) - \sum_{i=1}^{n} \log \left(1 + \sum_{l=1}^{K-1} e^{w_{l0} + w^T_l x_i}\right) - \frac{\rho}{2} w_{kj}^2. \quad (5)$$

Univariate Newton-Raphson algorithm

- $F_1$ and $F_2$ are smooth univariate concave functions in $w_{kj}$. Univariate Newton-Raphson algorithm can be used to update $w_{kj}$

$$w_{kj}^{(s+1)} = w_{kj}^{(s)} - \left( \frac{\partial^2 F(w_{kj}; \theta^{(q)})}{\partial^2 w_{kj}} \right)^{-1} \left|_{w_{kj}^{(s)}} \right. \left( \frac{\partial F(w_{kj}; \theta^{(q)})}{\partial w_{kj}} \right) - \gamma_k \text{sign}(w_{kj}) \right|_{w_{kj}^{(s)}},$$

where $\frac{\partial^2 F(w_{kj}; \theta^{(q)})}{\partial^2 w_{kj}}$ and $\frac{\partial F(w_{kj}; \theta^{(q)})}{\partial w_{kj}}$ have closed-form.
Updating the expert parameters

M-step (cont.)

- Update $\beta_{kj}$ using coordinate ascent algorithm with soft-thresholding operator

$$
\beta_{kj}^{[s+1]} = S_{\lambda_k \sigma_k} \left( \sum_{i=1}^{n} \tau_{ik}^{(q)} r_{ikj}^{[s]} x_{ij} \right) / \sum_{i=1}^{n} \tau_{ik}^{(q)} x_{ij},
$$

where $r_{ikj}^{[s]} = y_i - \beta_{k0}^{[s]} - \beta_{kj}^{[s]} x_i + \beta_{kj}^{[s]} x_{ij}$, $[S_{\gamma}(u)]_j = \text{sign}(u_j)(|u_j| - \gamma)_+$ and $(x)_+ = \max\{x, 0\}$ in the $s$th loop of the coordinate ascent algorithm.

$$
\beta_{k0}^{[s+1]} = \sum_{i=1}^{n} \tau_{ik}^{(q)} (y_i - x_i^\top \beta_{k}^{[s+1]}) / \sum_{i=1}^{n} \tau_{ik}^{(q)}.
$$
Updating the expert parameters

M-step (cont.)

- Update $\beta_{kj}$ using coordinate ascent algorithm with soft-thresholding operator

\[
\beta_{kj}^{[s+1]} = S_{\lambda_k} \sigma_k^{(q)2} \left( \sum_{i=1}^{n} \tau_{ik}^{(q)} r_{ikj}^{[s]} x_{ij} \right) / \sum_{i=1}^{n} \tau_{ik}^{(q)} x_{ij}^2,
\]

where $r_{ikj}^{[s]} = y_i - \beta_{kj}^{[s]} - \beta_k^{[s]T} x_i + \beta_{kj}^{[s]} x_{ij}$, $[S_{\gamma}(u)]_j = \text{sign}(u_j)(|u_j| - \gamma)_+$ and $(x)_+ = \max\{x, 0\}$ in the $s$th loop of the coordinate ascent algorithm.

\[
\beta_{k0}^{[s+1]} = \sum_{i=1}^{n} \tau_{ik}^{(q)} (y_i - x_i^T \beta_{kj}^{[s+1]}) / \sum_{i=1}^{n} \tau_{ik}^{(q)}.
\]

- Rerun the E-step, keep

\[
(w_{k0}^{(q+2)}, w_k^{(q+2)}) = (w_{k0}^{(q+1)}, w_k^{(q+1)}); (\beta_{k0}^{(q+2)}, \beta_k^{(q+2)}) = (\beta_{k0}^{(q+1)}, \beta_k^{(q+1)}),
\]

and update $\sigma_k^{2(q+2)}$ as follows

\[
\sigma_k^{2(q+2)} = \sum_{i=1}^{n} \tau_{ik}^{(q+1)} (y_i - \beta_{k0}^{(q+2)} - x_i^T \beta_{k}^{(q+2)})^2 / \sum_{i=1}^{n} \tau_{ik}^{(q+1)}.
\]
Simulation study

Simulation protocol

- $x \sim \mathcal{N}(0; \Sigma)$ with $\text{corr}(x_{ij}, x_{ij'}) = 0.5^{|j-j'|}; K = 2$
- Sample size: $n = 300$, 100 different data sets;
- The regression coefficients:
  \[
  (\beta_{10}, \beta_1)^T = (0, 0, 1.5, 0, 0, 0, 1)^T; \sigma_1 = 1
  
  (\beta_{20}, \beta_2)^T = (0, 1, -1.5, 0, 0, 2, 0)^T; \sigma_2 = 1
  
  (w_{10}, w_1)^T = (1, 2, 0, 0, -1, 0, 0)^T; \sigma_3 = 1
  \]

Considered approaches for comparison

- The standard MoE;
- MoE+$L_2$ (MoE with $L_2$ penalties in the gating network);
- MoE-BIC (MoE with model selection using BIC criterion - 100 submodels);
- MIXLASSO (MLR with Lasso penalties) (see Khalili and Chen (2007));

Evaluation criteria

- The sensitivity/specificity (sparsity);
- The parameter estimation (density estimation);
- The misclassification error: Adjust rand index - ARI (clustering).
Sensitivity/specificity result

- **Sensitivity** ($S_1$): proportion of correctly estimated zero coefficients;
- **Specificity**: proportion of correctly estimated nonzero coefficients.

<table>
<thead>
<tr>
<th>Method</th>
<th>Expert 1</th>
<th></th>
<th>Expert 2</th>
<th></th>
<th>Gate</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$S_1$</td>
<td>$S_2$</td>
<td>$S_1$</td>
<td>$S_2$</td>
<td>$S_1$</td>
<td>$S_2$</td>
</tr>
<tr>
<td>MoE</td>
<td>0.000</td>
<td>1.000</td>
<td>0.000</td>
<td>1.000</td>
<td>0.000</td>
<td>1.000</td>
</tr>
<tr>
<td>MoE+$L_2$</td>
<td>0.000</td>
<td>1.000</td>
<td>0.000</td>
<td>1.000</td>
<td>0.000</td>
<td>1.000</td>
</tr>
<tr>
<td>MoE-BIC</td>
<td>0.920</td>
<td>1.000</td>
<td>0.930</td>
<td>1.000</td>
<td>0.850</td>
<td>1.000</td>
</tr>
<tr>
<td>MIXLASSO</td>
<td>0.775</td>
<td>1.000</td>
<td>0.693</td>
<td>1.000</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td><strong>Our MoE-Lasso+$L_2$</strong></td>
<td>0.700</td>
<td>1.000</td>
<td>0.803</td>
<td>1.000</td>
<td>0.853</td>
<td>0.945</td>
</tr>
</tbody>
</table>

Table: Sensitivity ($S_1$) and specificity ($S_2$) results.

- MoE and MoE+$L_2$ could not be considered as model selection methods since their sensitivity equal zero.
- MIXLASSO can detect the zero coefficients in the experts. However, this model has a poor result when clustering the data.
- The MoE-Lasso+$L_2$ model can detect the zero coefficients in the experts and the gating network.
Parameter estimation for expert 1

\[(\beta_{10}, \beta_1)^T = (0, 0, 1.5, 0, 0, 0, 1)^T.\]
Parameter estimation for expert 2

- \((\beta_{20}, \beta_2)^T = (0, 1, -1.5, 0, 0, 2, 0)^T\).
Parameter estimation for gating network

- $(w_{10}, w_1)^T = (1, 2, 0, 0, -1, 0, 0)^T$. 

![Box plots for MoE, MoE-$L_2$, MoE-BIC, and MoE-Lasso + $L_2$.]
## Result for data clustering

<table>
<thead>
<tr>
<th>Model</th>
<th>MoE</th>
<th>MoE+$L_2$</th>
<th>MoE-BIC</th>
<th>MoE-Lasso + $L_2$</th>
<th>MIXLASSO</th>
</tr>
</thead>
<tbody>
<tr>
<td>C. rate</td>
<td>89.57% (1.65%)</td>
<td>89.62% (1.63%)</td>
<td>90.05% (1.65%)</td>
<td>89.46% (1.76%)</td>
<td>82.89% (1.92%)</td>
</tr>
<tr>
<td>ARI</td>
<td>0.6226 (.053)</td>
<td>0.6241 (.052)</td>
<td>0.6380 (.053)</td>
<td>0.6190 (.056)</td>
<td>0.4218 (.050)</td>
</tr>
</tbody>
</table>

Table: clustering accuracy results (correct classification rate and Adjusted Rand Index).

### Remarks

- MoE-BIC provides the best results. However, it is hard to apply BIC in reality especially for high dimensional data, since this involves a huge collection of model candidates.
- MIXLASSO can detect zero coefficients in the experts, but it provides a poor result when clustering data.
- MoE-Lasso+$L_2$ can detect zero coefficients in the model and provide a competitive result with MoE, MoE-$L_2$ in term of clustering, although it also causes bias to the non-zero coefficients.
Applications to real data sets

- For real data sets, we calculate the mean squared error and the correlation between the response variable $Y$ with its predictor $\hat{Y}$, where

$$\hat{Y} = \sum_{k=1}^{K} \pi_k(x; \hat{w})(\hat{\beta}_{k0} + x^T\hat{\beta}_k).$$

- Housing data: 13 features, 506 observations, $K = 2$.

<table>
<thead>
<tr>
<th></th>
<th>MoE</th>
<th>MoE-Lasso + $L_2$ (Khalili)</th>
<th>MoE-Lasso + $L_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$R^2$</td>
<td>0.8457</td>
<td>0.8094</td>
<td>0.8221</td>
</tr>
<tr>
<td>MSE</td>
<td>$0.1544_{(.577)}$</td>
<td>$0.2044_{(.709)}$</td>
<td>$0.1989_{(.619)}$</td>
</tr>
</tbody>
</table>

Table: Results for Housing data set.

- Baseball salary data: 32 features, 337 observations, $K = 2$.

<table>
<thead>
<tr>
<th></th>
<th>MoE</th>
<th>MoE-Lasso + $L_2$</th>
<th>MIXLASSO</th>
</tr>
</thead>
<tbody>
<tr>
<td>$R^2$</td>
<td>0.8099</td>
<td>0.8020</td>
<td>0.4252</td>
</tr>
<tr>
<td>MSE</td>
<td>$0.2625_{(.758)}$</td>
<td>$0.2821_{(.633)}$</td>
<td>$1.1858_{(2.792)}$</td>
</tr>
</tbody>
</table>

Table: Results for Baseball salaries data set.
The proximal Newton method

- We recently improve the proposed algorithm by using the proximal Newton method (Lee et al. (2006), Lee et al. (2014) and Friedman et al. (2010)) for updating the gating network parameters.

- The idea of the proximal Newton method:
  - Approximate the smooth part of $Q(w; \theta^{(q)})$ with its local quadratic form;
  - Use coordinate ascent with soft-thresholding operator to solve the resulting approximated convex optimization problem;
  - Combine with backtracking line search to update $w$. 
Extension result for proximal Newton method

- Coordinate ascent algorithm (CA) VS proximal Newton (PN) method:

<table>
<thead>
<tr>
<th>Criteria</th>
<th>MoE-Lasso + $L_2$ (CA)</th>
<th>MoE-Lasso + $L_2$ (PN)</th>
</tr>
</thead>
<tbody>
<tr>
<td>C.Rate</td>
<td>89.46% (1.76%)</td>
<td>89.53% (1.65%)</td>
</tr>
<tr>
<td>ARI</td>
<td>0.6190 (.056)</td>
<td>0.6210 (.052)</td>
</tr>
<tr>
<td>$PL(\theta)$ value</td>
<td>$-558.140 (12.99)$</td>
<td>$-558.410 (13.03)$</td>
</tr>
</tbody>
</table>

Table: Simulation results.

- Application of the proximal Newton algorithm to the residential building data set: 107 features, 372 observations, $K = 3$.

<table>
<thead>
<tr>
<th>Method</th>
<th>Before clustering</th>
<th>After clustering</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$R^2$</td>
<td>MSE</td>
</tr>
<tr>
<td>Proximal Newton</td>
<td>0.9887</td>
<td>0.0120 (.879)</td>
</tr>
</tbody>
</table>

Table: Results for residential building data set.
Conclusion and perspectives

Conclusion

- We propose a regularized MoE which does not require using approximations as in standard MoE regularization.
- A blockwise EM algorithm with coordinate ascent algorithm is proposed to monotonically maximize the RMoE objective function.
- The updating of the gating network for some situations is time consuming since we don’t have a closed-form.
- The algorithm has been improved by using proximal Newton method to update the gating network, which has a closed-form update for each parameter and improve the running time.
- Future work: Estimation and feature selection for hierarchical MoE and MoE with discrete data, ...
- Consider the case \( p \gg n \).
Thank you!


