機器學習分類方法於潛在類別迴歸模型的超參數選擇

冼航平、黃冠華
交通大學統計學研究所

第二十八屆南區統計研討會
2019/6/21 台中中興大學

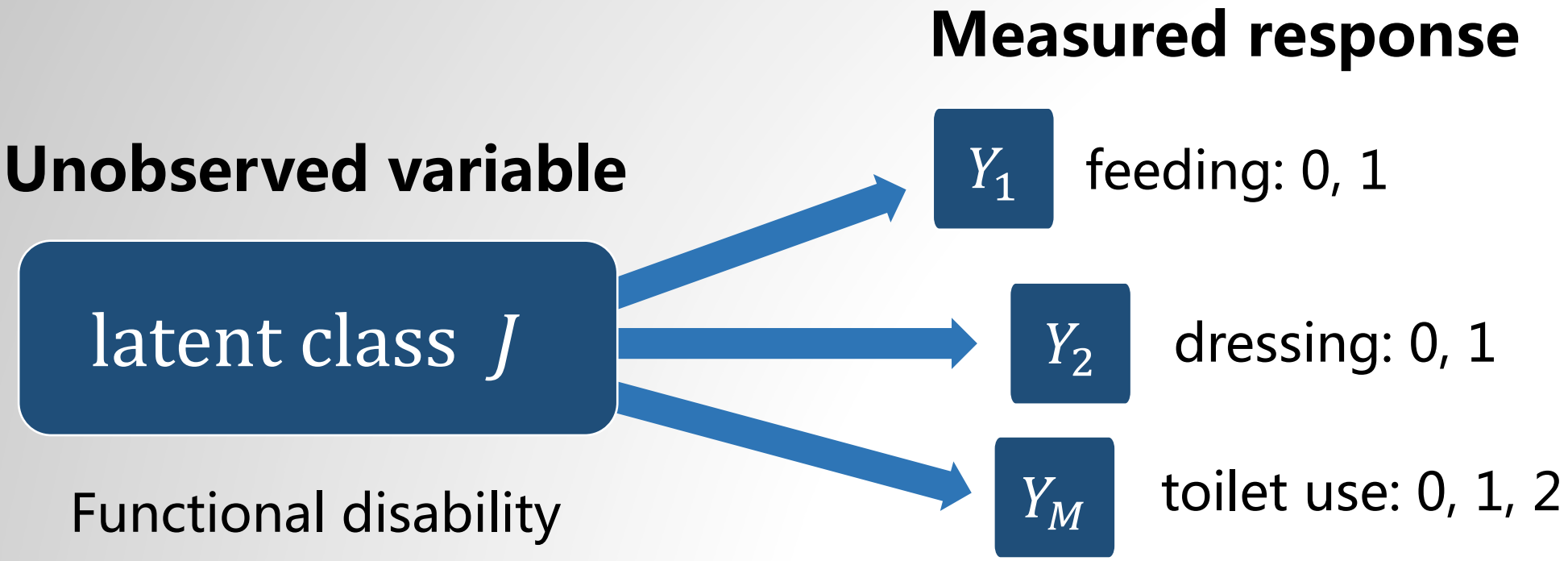
Contents

大綱

研究背景

Background

Latent Class Analysis (LCA) :



LCA model :

$$\Pr(Y_{i1} = y_1, \ldots, Y_{iM} = y_m) = \sum_{j=1}^{J} \left\{ \eta_j \prod_{m=1}^{M} \prod_{k=1}^{K_m} p_{mkj}^{y_{mk}} \right\}$$

where $\eta_j = \Pr(S_i = j)\,,\ j = 1 \ldots J$

$p_{mkj} = \Pr(Y_{im} = k | S_i = j),\ y_{mk} = \mathrm{I}(y_m = k),\ m = 1, \ldots, M,\ k = 1, \ldots, K_m$

Assumptions :

i. Both latent variable and measured responses are discrete.

ii. $\Pr(Y_{i1} = y_1, \ldots, Y_{iM} = y_m | S_i) = \prod_{m=1}^{M} \Pr(Y_{im} = y_m | S_i)$

LCR (latent class regression) model :

$\eta_j \rightarrow \eta_j(\mathbf{x}_i)$ covariate effects on latent prevalence

- Dayton & Macready, 1988
- Van der Heijden, Dessens, & Böckenholt, 1996
- Bandeen-Roche, Miglioreti, Zeger, & Rathouz, 1997

$p_{mkj} \rightarrow p_{mkj}(\mathbf{z}_{im})$ covariate effects on conditional probabilities

- Melton, Liang, & Pulver, 1994

We use the identifiable LCR model incorporating covariates to predict both the latent and measured outcomes.

- Huang & Bandeen-Roche, 2004

LCR (latent class regression) model :

$$\Pr(Y_{i1} = y_1, \ldots, Y_{iM} = y_m) = \sum_{j=1}^{J} \left\{ \eta_j(\mathbf{x}_i) \prod_{m=1}^{M} \prod_{k=1}^{K_m} p_{mkj}^{y_{mk}}(\mathbf{z}_{im}) \right\}$$

$\mathbf{x}_i = (1, x_{i1}, \ldots, x_{ip})^T$

$\mathbf{z}_i = (\mathbf{z}_{i1}, \ldots, \mathbf{z}_{iM})$ with $\mathbf{z}_{im} = (1, z_{im1}, \ldots, z_{imL})^T$

- The covariates may include any combination of continuous and discrete measure.
- Two sets of covariates can be mutually exclusive or overlap.

LCR (latent class regression) model :

$$\Pr(Y_{i1} = y_1, \ldots, Y_{iM} = y_m) = \sum_{j=1}^{J} \left\{ \eta_j(\mathbf{x}_i) \prod_{m=1}^{M} \prod_{k=1}^{K_m} p_{mkj}^{y_{mk}}(\mathbf{z}_{im}) \right\}$$

$\eta_j(\mathbf{x}_i)$, $p_{mkj}(\mathbf{z}_{im})$ are related to $\mathbf{x}_i$, $\mathbf{z}_{im}$ via generalized logit link functions :

$$\log\left[\frac{\eta_j(\mathbf{x}_i)}{\eta_J(\mathbf{x}_i)}\right] = \beta_{0j} + \beta_{1j}x_{i1} + \cdots + \beta_{Pj}x_{iP} = \mathbf{x}_i^T \boldsymbol{\beta}_j$$

with $i = 1, \ldots, N;\ \ j = 1, \ldots, J-1$

$$\log\left[\frac{p_{mkj'}(\mathbf{z}_{im})}{p_{mK_mj'}(\mathbf{z}_{im})}\right] = \gamma_{mkj'} + \alpha_{1mk}z_{im1} + \cdots + \alpha_{Lmk}z_{imL} = \gamma_{mkj'} + \mathbf{z}_{im}^T \boldsymbol{\alpha}_{mk}$$

with $i = 1, \ldots, N;\ \ j' = 1, \ldots, J;\ \ m = 1, \ldots, M;\ \ k = 1, \ldots, K_m - 1$

LCR (latent class regression) model :

$$\Pr(Y_{i1} = y_1, \ldots, Y_{iM} = y_m) = \sum_{j=1}^{J} \left\{ \eta_j(\mathbf{x}_i) \prod_{m=1}^{M} \prod_{k=1}^{K_m} p_{mkj}^{y_{mk}}(\mathbf{z}_{im}) \right\}$$

$$\log\left[\frac{p_{mkj'}(\mathbf{z}_{im})}{p_{mK_mj'}(\mathbf{z}_{im})}\right] = \gamma_{mkj'} + \alpha_{1mk} z_{im1} + \cdots + \alpha_{Lmk} z_{imL} = \gamma_{mkj'} + \mathbf{z}_{im}^T \boldsymbol{\alpha}_{mk}$$

with $i = 1, \ldots, N;\ j' = 1, \ldots, J;\ m = 1, \ldots, M;\ k = 1, \ldots, K_m - 1$

- By fixing $\gamma_{mkj'}$ at positive or negative infinity, we can fit a constrained LCR model with the corresponding conditional probabilities being 1 or 0.

LCR (latent class regression) model :

$$\Pr(Y_{i1} = y_1, \ldots, Y_{iM} = y_m) = \sum_{j=1}^{J} \left\{ \eta_j(\mathbf{x}_i) \prod_{m=1}^{M} \prod_{k=1}^{K_m} p_{mkj}^{y_{mk}}(\mathbf{z}_{im}) \right\}$$

Assumptions :

i. $\Pr(S_i = j|\mathbf{x}_i, \mathbf{z}_i) = \Pr(S_i = j|\mathbf{x}_i)$

ii. $\Pr(Y_{i1} = y_1, \ldots, Y_{iM} = y_m|S_i, \mathbf{x}_i, \mathbf{z}_i) = \Pr(Y_{i1} = y_1, \ldots, Y_{iM} = y_m|S_i, \mathbf{z}_i)$

iii. $\Pr(Y_{i1} = y_1, \ldots, Y_{iM} = y_m|S_i, \mathbf{z}_i) = \prod_{m=1}^{M} \Pr(Y_{im} = y_m|S_i, \mathbf{z}_{im})$

Problems : $\eta_j \rightarrow \eta_j(\mathbf{x}_i),\ p_{mkj} \rightarrow p_{mkj}(\mathbf{z}_{im})$

- We can adjust for characteristics that determine measured responses other than underlying classes, hence hopefully improving the accuracy of classifying individuals.

- Sometimes, there many combinations of $\mathbf{x}_i$ and $\mathbf{z}_{im}$, and we do not have idea to choose which combination should be used to fit LCR model.

- We develop a method based on machine learning classification technique to select the optimal hyper-parameters.

Problems : $\eta_j \rightarrow \eta_j(\mathbf{x}_i),\ p_{mkj} \rightarrow p_{mkj}(\mathbf{z}_{im})$

Hyper-parameters :

1. Number of latent class, J
2. covariate of latent class prevalence, $\mathbf{x}_i$
3. covariate of conditional probabilities of measured response, $\mathbf{z}_{im}$

研究方法

Methodology

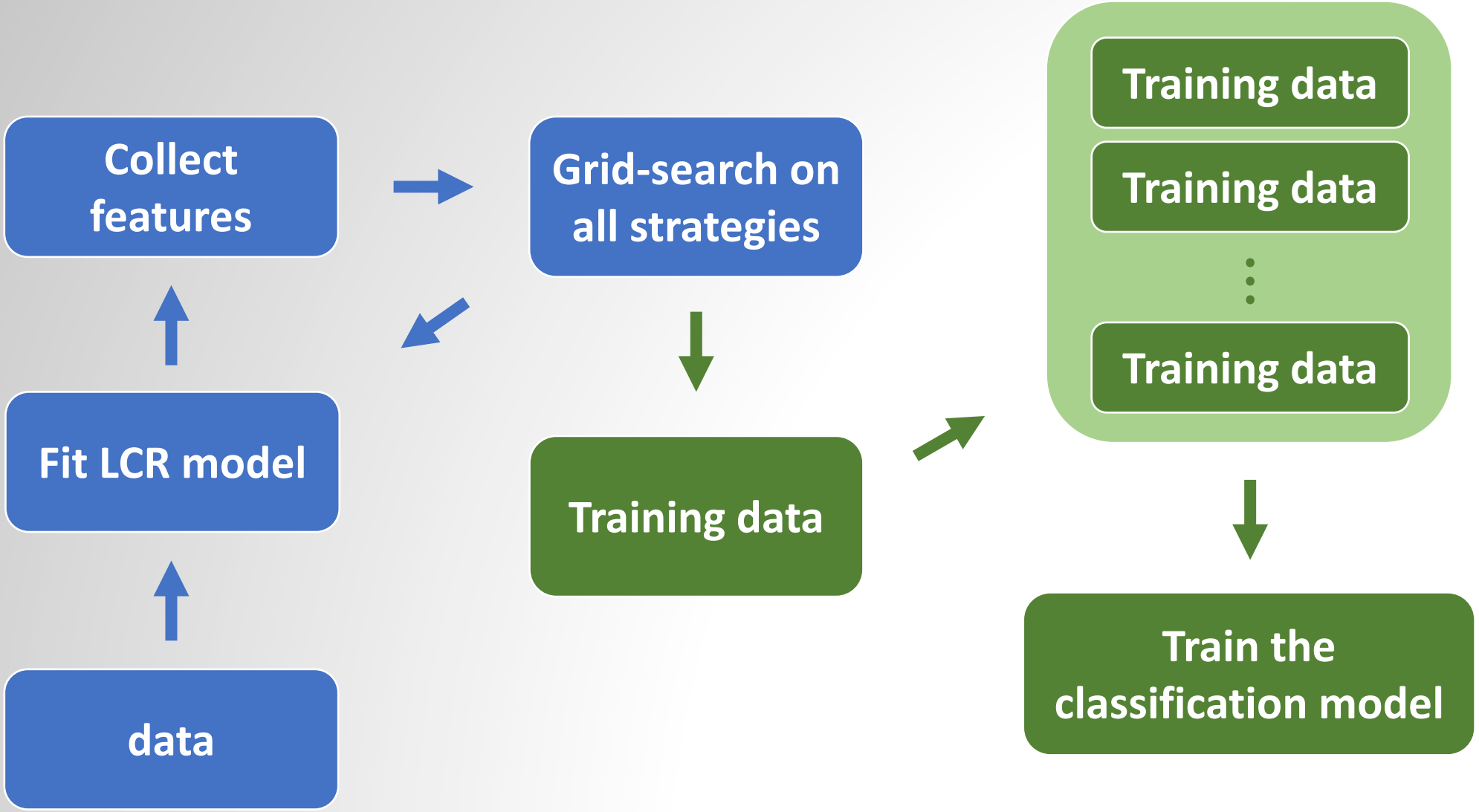
Brief introduction for our methodology

1. We want to build a **classification model** to select optimal LCR model's hyper-parameters.

2. We first define several strategies to modify hyper-parameters. **These strategies are our classification model's labels**.

3. When a LCR model is fitted, we use several **diagnostic statistics as features** to classify which strategy should be implemented.

How to build the classification model



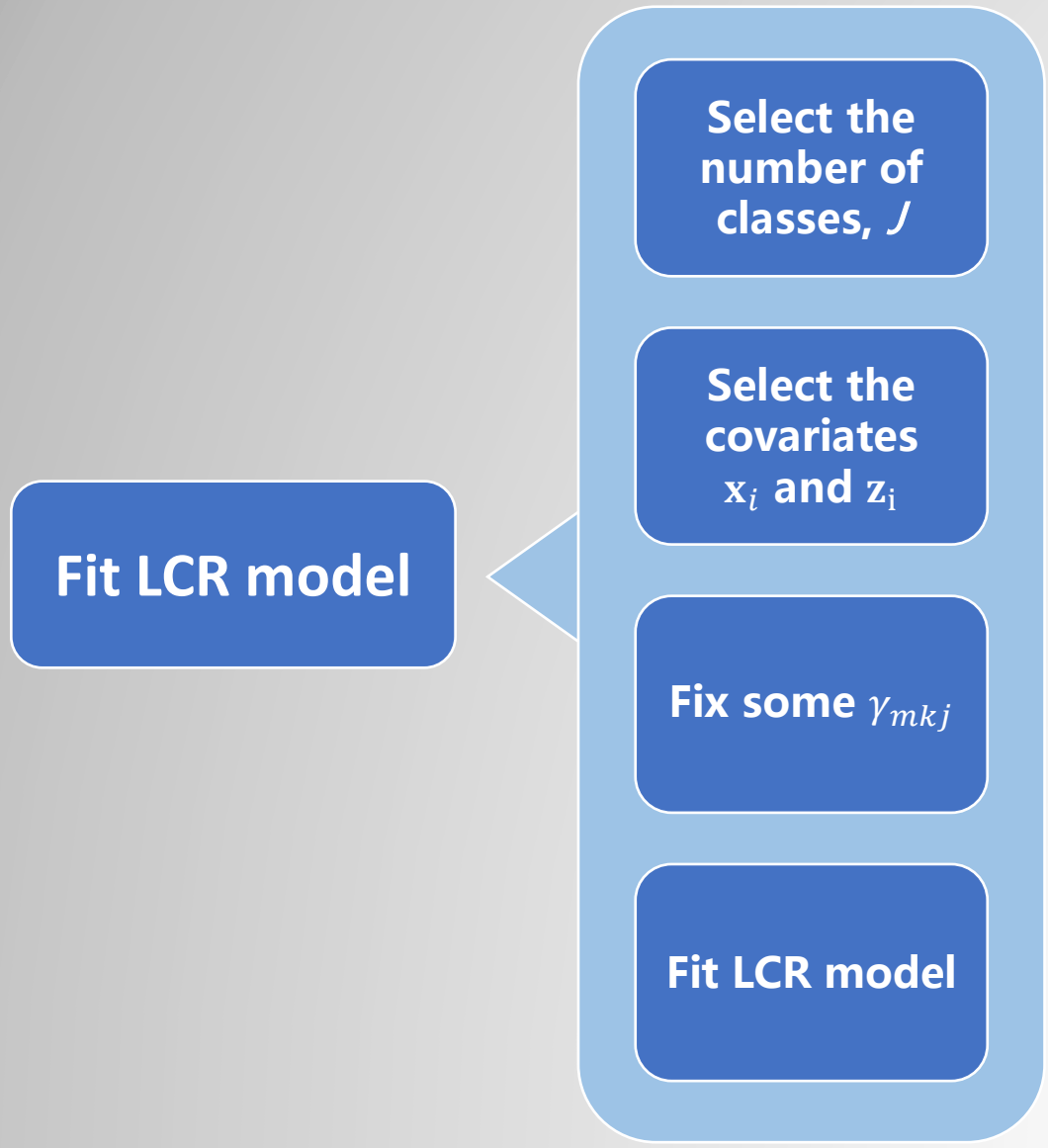
How to build the classification model

Several datasets used to build training dataset :

1. the data in Table 1 of Goodman (1974)
2. 8 simulation datasets in Huang (2005)
3. functional trajectory data in Chen (2008)
4. PANSS data in Huang (2011)
5. mHELP data in Chen (2014)

- **Due to the cost of computation time, the data all above have at most three levels in measured responses.**

How to build the classification model



data

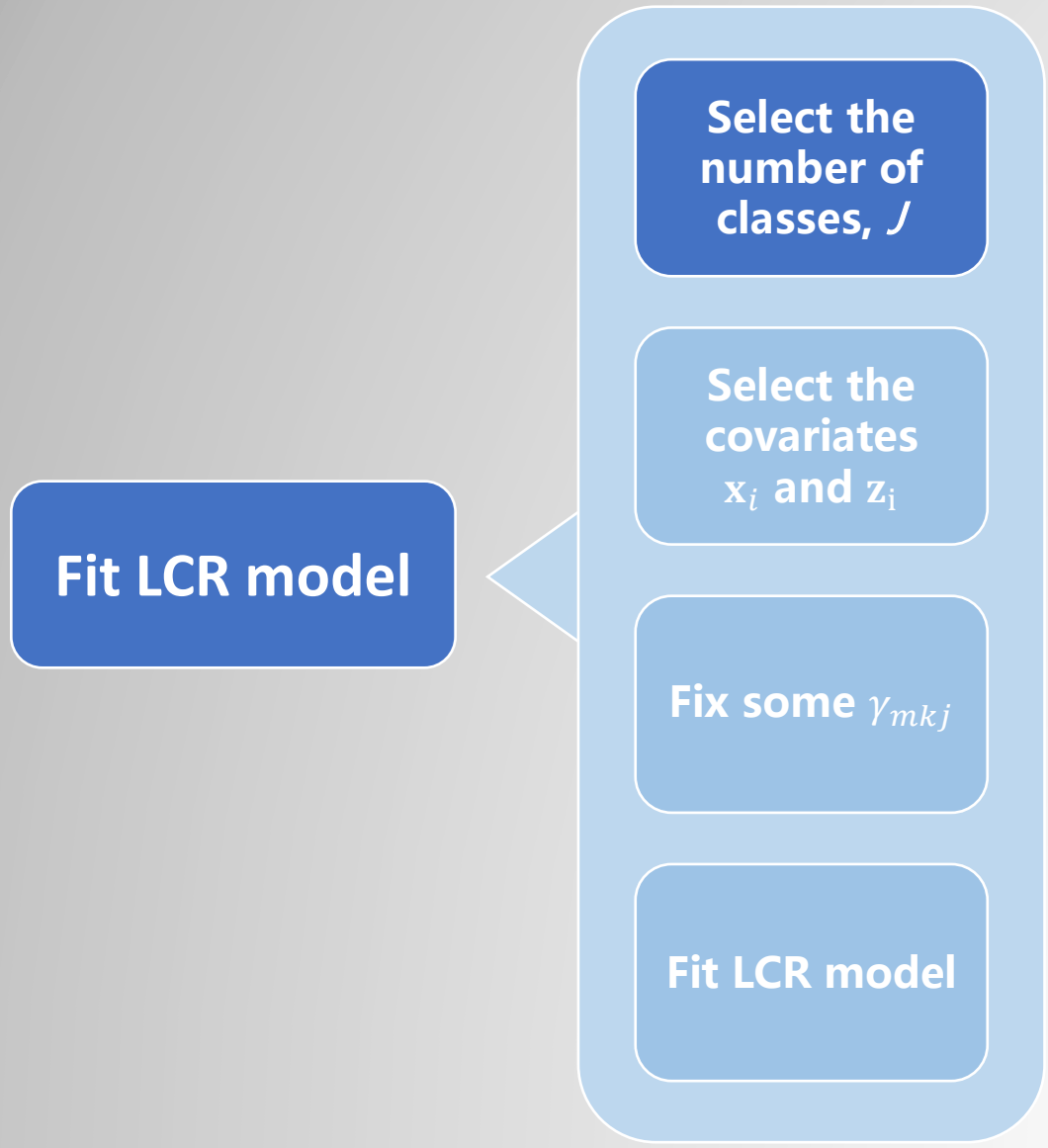
Fit LCR model

Collect features

Grid-search on all strategies

Train the classification model

How to build the classification model



1. Select J based on prior knowledge or the scientific objective.

2. The eigenvalue criterion of selecting the number of classes proposed by Huang (2005).

data

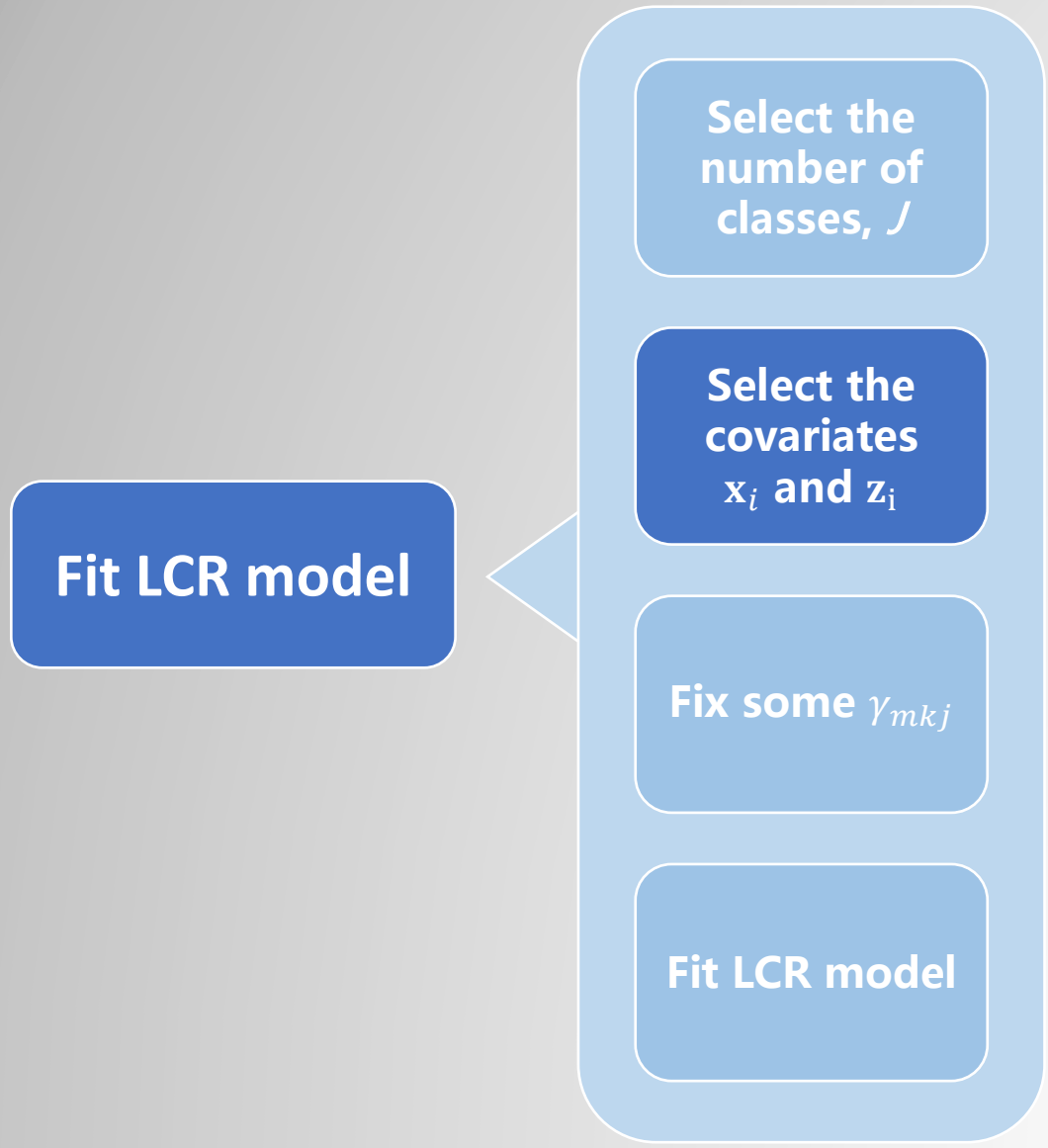
Fit LCR model

Collect features

Grid-search on all strategies

Train the classification model

How to build the classification model



Choose the relevant covariates as much as possible.

data

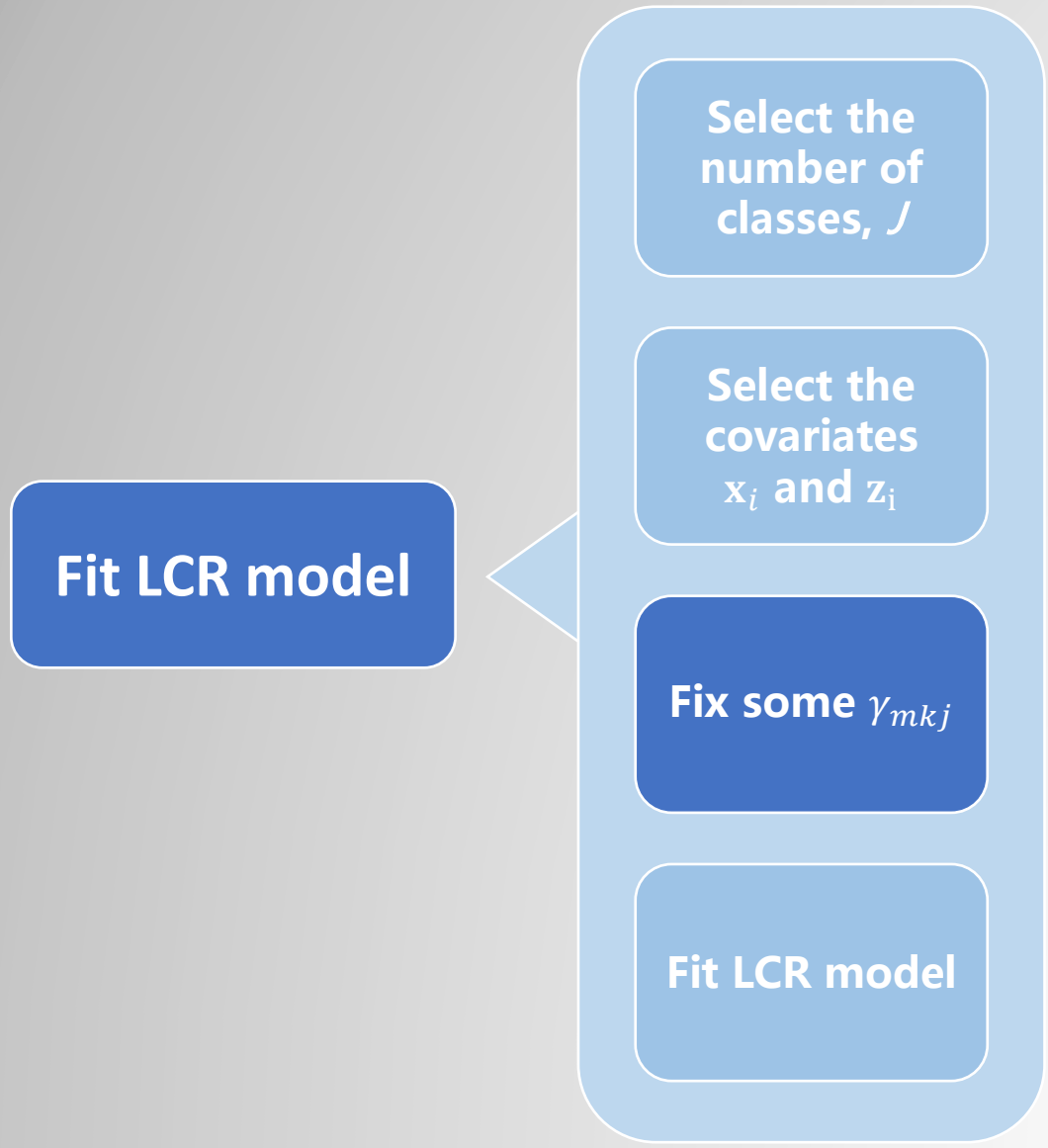
Fit LCR model

Collect features

Grid-search on all strategies

Train the classification model

How to build the classification model

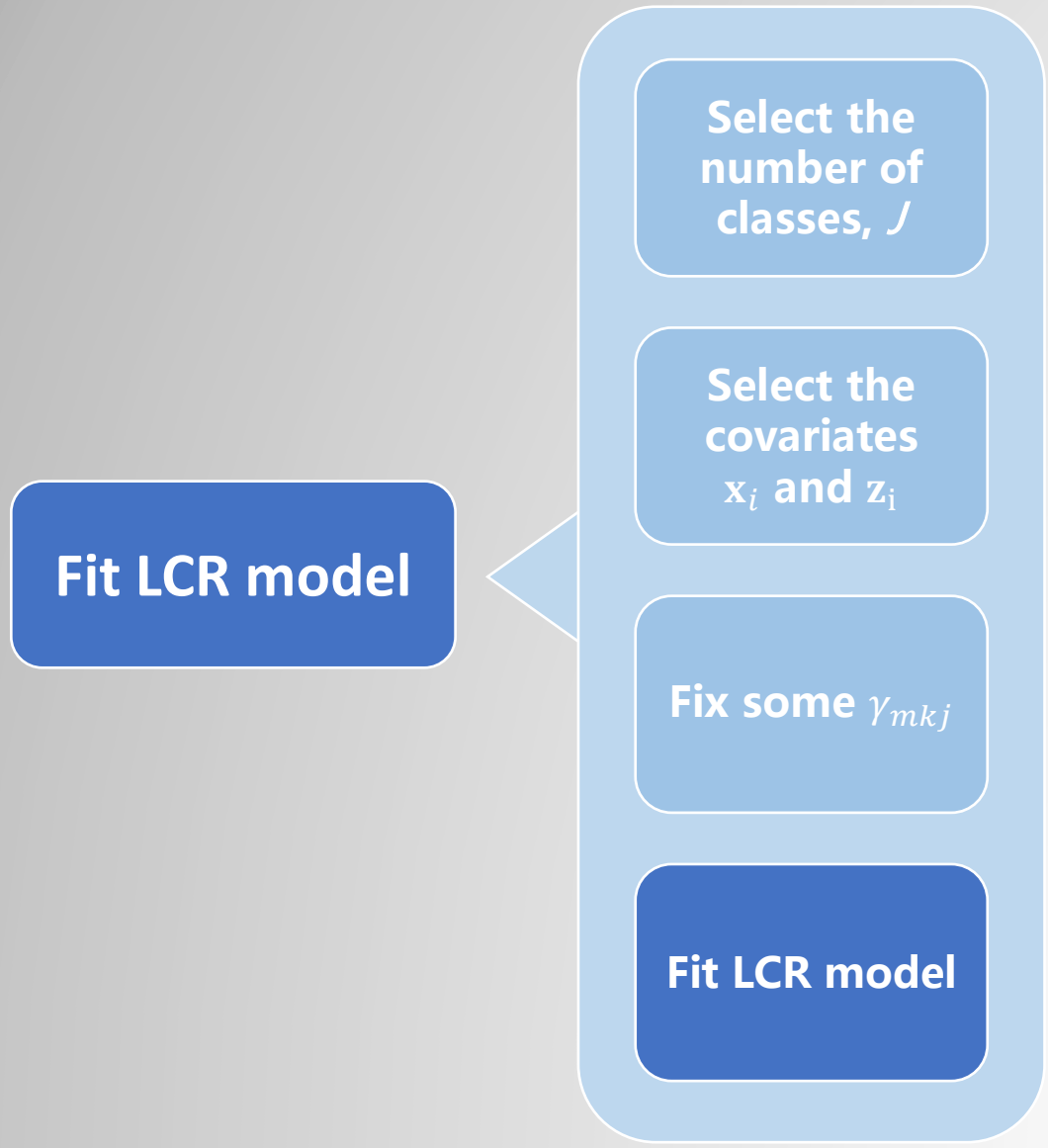


1. **Set according to the prior knowledge.**

2. **Fit model without fixed γ_{mkj} first, then fix γ_{mkj} with the largest standard errors.**

Repeat this procedure until all standard errors are less than $Q_3 + 1.5 \times IQR$

How to build the classification model



Use Expectation-Maximization (EM) algorithm introduced in Huang (2004) to estimate the parameters and their standard errors.

data

Fit LCR model

Collect features

Grid-search on all strategies

Train the classification model

How to build the classification model

3 types of 12 diagnostic statistics in total are used as the input features for our machine learning classifier :

Overall goodness-of fit :

$X^2_{LCA}.p$ $D^2_{LCA}.p$ $X^2_{LCR}.p$ $D^2_{LCR}.p$ $G^2.p$

Model Checking Based on the Pseudo-class Membership :

$X^2_{class}.p$ $D^2_{class}.p$ $\overline{X^2_{cp}.p}$ $\overline{D^2_{cp}.p}$ LoI

Identifiability for LCA/LCR model :

I_{LCA} I_{LCR}

data

Fit LCR model

Collect features

Grid-search on all strategies

Train the classification model

How to build the classification model

$X^2_{LCA}.p$ $D^2_{LCA}.p$ $X^2_{LCR}.p$ $D^2_{LCR}.p$ $G^2.p$ $X^2_{class}.p$ $D^2_{class}.p$ $\overline{X^2_{cp}.p}$ $\overline{D^2_{cp}.p}$ LoI I_{LCA} I_{LCR}

Overall goodness-of fit :

$X^2_{LCA}.p$ $D^2_{LCA}.p$ $X^2_{LCR}.p$ $D^2_{LCR}.p$ $G^2.p$

- We can think of the relationships among *M* polytomous variables as a *M*-way contingency table.

data

Fit LCR model

Collect features

Grid-search on all strategies

Train the classification model

How to build the classification model

$X^2_{LCA}.p$	$D^2_{LCA}.p$	$X^2_{LCR}.p$	$D^2_{LCR}.p$	$G^2.p$	$X^2_{class}.p$	$D^2_{class}.p$	$\overline{X^2_{cp}.p}$	$\overline{D^2_{cp}.p}$	LoI	I_{LCA}	I_{LCR}

Pearson Residuals and Deviance Residuals of LCA models :

Let

- $K^{\#} = \prod_{m=1}^{M} K_m$, $K^{*} = \sum_{m=1}^{M} K_m$
- ϕ be the vector of model parameters
- For $h = 1, \dots, K^{\#}$, let $\mathbf{y}_h = (y_{h1}, \dots, y_{hM})^T$ be hth possible response pattern of $\mathbf{Y}_i = (Y_{i1}, \dots, Y_{iM})^T$
- $\tilde{Y}_{ih} = \mathrm{I}(\mathbf{Y}_i = \boldsymbol{y}_{\mathrm{h}})$ be the indicator of whether the ith subject has the hth response
- $\pi_{ih}(\phi) = \Pr(\mathbf{Y}_i = \boldsymbol{y}_{\mathrm{h}}; \phi)$

Then the distribution of the ith subject in terms of $\tilde{Y}_{ih}$ can be expressed as :

$$\Pr(\mathbf{Y}_i = \boldsymbol{y}) = \Pr\left(\tilde{Y}_{i1} = \tilde{y}_1, \dots, \tilde{Y}_{iK^{\#}} = \tilde{y}_{K^{\#}}\right) = \prod_{h=1}^{K^{\#}} [\pi_{ih}(\phi)]^{\tilde{Y}_{ih}}$$

data

Fit LCR model

Collect features

Grid-search on all strategies

Train the classification model

How to build the classification model

$X^2_{LCA}.p$	$D^2_{LCA}.p$	$X^2_{LCR}.p$	$D^2_{LCR}.p$	$G^2.p$	$X^2_{class}.p$	$D^2_{class}.p$	$\overline{X^2_{cp}.p}$	$\overline{D^2_{cp}.p}$	LoI	I_{LCA}	I_{LCR}

Pearson Residuals and Deviance Residuals of LCA models :

Then the goodness-of-fit statistic and its p-value are :

$$X^2_{LCA} = \sum_{h=1}^{K^\#} \frac{(f_h - \hat{f}_h)^2}{\hat{f}_h}\ ,\ X^2_{LCA}.p = \text{p} - \text{value of } X^2_{LCA}$$

the corresponding likelihood ratio statistic and its p-value are :

$$D^2_{LCA} = 2\sum_{h=1}^{K^\#} f_h \log\left(\frac{f_h}{\hat{f}_h}\right)\ ,\ D^2_{LCA}.p = \text{p} - \text{value of } D^2_{LCA}$$

with d.f. $\left(K^\# - 1\right) - \left(J(K^* - M) + J - 1\right)$

where $f_h = \sum_{i=1}^{N} \tilde{Y}_{ih}$, $\hat{f}_h = \sum_{i=1}^{N} \pi_{ih}(\hat{\phi}) = N \times \pi_h(\hat{\phi}_{LCA})$ with $\phi_{LCA} = (\eta_j, p_{mkj})$

data

Fit LCR model

Collect features

Grid-search on all strategies

Train the classification model

How to build the classification model

$X^2_{LCA}.p$	$D^2_{LCA}.p$	$X^2_{LCR}.p$	$D^2_{LCR}.p$	$G^2.p$	$X^2_{class}.p$	$D^2_{class}.p$	$\overline{X^2_{cp}.p}$	$\overline{D^2_{cp}.p}$	LoI	I_{LCA}	I_{LCR}

Pearson Residuals and Deviance Residuals of LCR models :

Then the goodness-of-fit statistic and its pseudo p-value are :

$$X^2_{LCR} = \sum_{h=1}^{K^\#} \frac{(f_h - \hat{f}_h)^2}{\hat{f}_h}\ ,\ X^2_{LCR}.p = \text{p} - \text{value of } X^2_{LCR}$$

the corresponding likelihood ratio statistic and its pseudo p-value are :

$$D^2_{LCR} = 2\sum_{h=1}^{K^\#} f_h \log\left(\frac{f_h}{\hat{f}_h}\right)\ ,\ D^2_{LCR}.p = \text{p} - \text{value of } D^2_{LCR}$$

where $f_h = \sum_{i=1}^{N} \tilde{Y}_{ih}$, $\hat{f}_h = \pi_{ih}(\hat{\phi}_{LCR})$ with $\phi_{LCR} = (\boldsymbol{\beta}_j, \gamma_{mkj'}, \boldsymbol{\alpha}_{mk})$

data

Fit LCR model

Collect features

Grid-search on all strategies

Train the classification model

How to build the classification model

$X^2_{LCA}.p$	$D^2_{LCA}.p$	$X^2_{LCR}.p$	$D^2_{LCR}.p$	$G^2.p$	$X^2_{class}.p$	$D^2_{class}.p$	$\overline{X^2_{cp}.p}$	$\overline{D^2_{cp}.p}$	LoI	I_{LCA}	I_{LCR}

G^2 statistic to test the goodness of fit for LCR model :

Consider $\tilde{\mathbf{Y}}_i = \left(\tilde{Y}_{i1}, \ldots, \tilde{Y}_{i(K^{\#}-1)}\right)^T$

Under the LCR model, $\tilde{\mathbf{Y}}_i \sim multinomial\left(1; \pi_{i1}(\phi), \ldots, \pi_{i(K^{\#}-1)}(\phi)\right)$

with

$$\mathbf{V}_i = \mathrm{Var}(\tilde{\mathbf{Y}}_i) = \begin{bmatrix} \pi_{i1}(\phi)(1-\pi_{i1}(\phi)) & -\pi_{i1}(\phi)\pi_{i2}(\phi) & \cdots & -\pi_{i1}(\phi)\pi_{i(K^{\#}-1)}(\phi) \\ -\pi_{i2}(\phi)\pi_{i1}(\phi) & \pi_{i2}(\phi)(1-\pi_{i2}(\phi)) & \cdots & -\pi_{i2}(\phi)\pi_{i(K^{\#}-1)}(\phi) \\ \vdots & \vdots & \ddots & \vdots \\ -\pi_{i(K^{\#}-1)}(\phi)\pi_{i1}(\phi) & -\pi_{i(K^{\#}-1)}(\phi)\pi_{i2}(\phi) & \cdots & \pi_{i(K^{\#}-1)}(\phi)\left(1-\pi_{i(K^{\#}-1)}(\phi)\right) \end{bmatrix}$$

then we can obtain the following theorem:

data

Fit LCR model

Collect features

Grid-search on all strategies

Train the classification model

How to build the classification model

$X^2_{LCA}.p$	$D^2_{LCA}.p$	$X^2_{LCR}.p$	$D^2_{LCR}.p$	$G^2.p$	$X^2_{class}.p$	$D^2_{class}.p$	$\overline{X^2_{cp}.p}$	$\overline{D^2_{cp}.p}$	LoI	I_{LCA}	I_{LCR}

G^2 statistic to test the goodness of fit for LCR model :

Theorem. *Let* $\hat{\mathbf{S}}_n = \frac{1}{\sqrt{n}}\sum_{i=1}^{n}\left(\tilde{\mathbf{Y}}_i - \boldsymbol{\pi}_i(\hat{\phi}_n)\right)$, *where* $\boldsymbol{\pi}_i(\hat{\phi}_n) = \left(\pi_{i1}(\hat{\phi}_n), \ldots, \pi_{i(K^{\#}-1)}(\hat{\phi}_n)\right)^T$ *and* $\hat{\phi}_n$ *is the estimate of* ϕ *from* $\{\tilde{\mathbf{Y}}_i; i = 1, \ldots, n\}$. *Suppose that*

(i) *all elements of* $\phi, \mathbf{x}_i$ *and* $\mathbf{z}_i, i = 1, \ldots, n$ *are finite;*

(ii) $\boldsymbol{\Sigma}_n = \frac{1}{n}\sum_{i=1}^{n}\mathbf{V}_i \rightarrow \boldsymbol{\Sigma}$ *which is positive definite;*

(iii) *there exist postive numbers* A *and* B *such that for all* i, p, m *and* q, $|x_{ip}| \leq A$ *and* $|z_{imq}| \leq B$.

Then $G^2 = \hat{\mathbf{S}}_n^T \boldsymbol{\Sigma}^{-1} \hat{\mathbf{S}}_n \xrightarrow{\mathcal{L}} \chi^2$ *with* $d.f. = (K^{\#} - 1)$

$G^2.p = p - value\ of\ G^2$

data

Fit LCR model

Collect features

Grid-search on all strategies

Train the classification model

How to build the classification model

$X^2_{LCA}.p$ | $D^2_{LCA}.p$ | $X^2_{LCR}.p$ | $D^2_{LCR}.p$ | $G^2.p$ | $X^2_{class}.p$ | $D^2_{class}.p$ | $\overline{X^2_{cp}.p}$ | $\overline{D^2_{cp}.p}$ | LoI | I_{LCA} | I_{LCR}

Model Checking Based on the Pseudo-class Membership :

$X^2_{class}.p$ | $D^2_{class}.p$ | $\overline{X^2_{cp}.p}$ | $\overline{D^2_{cp}.p}$ | LoI

After $(\mathbf{Y}_i, \mathbf{x}_i, \mathbf{z}_i)$ is observed for subject i, the posterior probabilities of class membership is :

$$\theta_{ij} = \Pr(S_i = j | \mathbf{Y}_i = y_i, \mathrm{x}_i, z_i) = \frac{\eta_j(\mathbf{x}_i) \prod_{m=1}^{M} \prod_{k=1}^{K_m} [p_{mkj}(\mathbf{z}_{im})]^{y_{imk}}}{\sum_{l=1}^{J} \eta_l(\mathbf{x}_i) \prod_{m=1}^{M} \prod_{k=1}^{K_m} [p_{mkl}(\mathbf{z}_{im})]^{y_{imk}}}, \qquad j = 1, \ldots, J$$

we can get subject i's pseudo-class membership by randomly assigning it to a stratum $C_i \in \{1, \ldots, J\}$, with probabilities $\hat{\theta}_{i1}, \ldots, \hat{\theta}_{iJ}$

data

Fit LCR model

Collect features

Grid-search on all strategies

Train the classification model

How to build the classification model

$X^2_{LCA}.p$	$D^2_{LCA}.p$	$X^2_{LCR}.p$	$D^2_{LCR}.p$	$G^2.p$	$X^2_{class}.p$	$D^2_{class}.p$	$\overline{X^2_{cp}.p}$	$\overline{D^2_{cp}.p}$	LoI	I_{LCA}	I_{LCR}

Goodness of fit for diagnosis of latent prevalence $\eta_j(\mathbf{x}_i)$:

Define $\tilde{C}_{ij} = \mathrm{I}(C_i = j)$, $u_j = \sum_{i=1}^{N} \tilde{C}_{ij}$ and $\hat{u}_j = \sum_{i=1}^{N} \hat{\eta}_j(\mathbf{x}_i)$

Then the goodness-of-fit statistic and its p-value are :

$$X^2_{class} = \sum_{j=1}^{J} \frac{(u_j - \hat{u}_j)^2}{\hat{u}_j}, \quad X^2_{class}.p = \mathrm{p-value\ of}\ X^2_{class}$$

the corresponding likelihood ratio statistic and its p-value are :

$$D^2_{class} = 2\sum_{j=1}^{J} u_j \log\left(\frac{u_j}{\hat{u}_j}\right), \quad D^2_{class}.p = \mathrm{p-value\ of}\ D^2_{class}$$

with d.f. $J - 1$

data

Fit LCR model

Collect features

Grid-search on all strategies

Train the classification model

How to build the classification model

$X^2_{LCA}.p$	$D^2_{LCA}.p$	$X^2_{LCR}.p$	$D^2_{LCR}.p$	$G^2.p$	$X^2_{class}.p$	$D^2_{class}.p$	$\overline{X^2_{cp}.p}$	$\overline{D^2_{cp}.p}$	LoI	I_{LCA}	I_{LCR}

Goodness of fit for diagnosis of conditional probabilities $p_{mkj}(\mathbf{z}_{im})$:

Define $\tilde{Y}_{imkj} = \mathrm{I}(Y_{im} = k \cap C_i = j)$,
$o_{mkj} = \sum_{i=1}^{N} \tilde{Y}_{imkj}$ and $\hat{o}_{mkj} = \sum_{i=1}^{N} \hat{\eta}_j(\mathbf{x}_i)\hat{p}_{mkj}(\mathbf{z}_{im})$

Then the goodness-of-fit statistic and its p-value are :

$$X^2_{mj} = \sum_{k=1}^{K_m} \frac{(o_{mkj} - \hat{o}_{mkj})^2}{\hat{o}_{mkj}}$$

the corresponding likelihood ratio statistic and its p-value are :

$$D^2_{mj} = 2\sum_{k=1}^{K_m} o_{mkj} \log\left(\frac{o_{mkj}}{\hat{o}_{mkj}}\right)$$

with d.f. $K_m - 1$

data

Fit LCR model

Collect features

Grid-search on all strategies

Train the classification model

How to build the classification model

$X^2_{LCA}.p$	$D^2_{LCA}.p$	$X^2_{LCR}.p$	$D^2_{LCR}.p$	$G^2.p$	$X^2_{class}.p$	$D^2_{class}.p$	$\overline{X^2_{cp}.p}$	$\overline{D^2_{cp}.p}$	LoI	I_{LCA}	I_{LCR}

Goodness of fit for diagnosis of conditional probabilities $p_{mkj}(\mathbf{z}_{im})$:

We use the weighted average of these two statistics as the diagnosis statistics of LCR model :

$$\overline{X^2_{cp}} = \frac{1}{M}\sum_{m=1}^{M}\sum_{j=1}^{J} w_j X^2_{mj}$$

and

$$\overline{D^2_{cp}} = \frac{1}{M}\sum_{m=1}^{M}\sum_{j=1}^{J} w_j D^2_{mj}$$

where $w_j =$ (the number of objects in class j)$/N$

d.f. can be obtained by the same weighted averaging approach over the p-values of X^2_{mj} and D^2_{mj}, respectively.

data

Fit LCR model

Collect features

Grid-search on all strategies

Train the classification model

How to build the classification model

$X^2_{LCA}.p$	$D^2_{LCA}.p$	$X^2_{LCR}.p$	$D^2_{LCR}.p$	$G^2.p$	$X^2_{class}.p$	$D^2_{class}.p$	$\overline{X^2_{cp}.p}$	$\overline{D^2_{cp}.p}$	LoI	I_{LCA}	I_{LCR}

Check the conditional independence assumption :

$$\Pr(Y_{i1} = y_1, \ldots, Y_{iM} = y_m | S_i, \mathbf{z}_i) = \prod_{m=1}^{M} \Pr(Y_{im} = y_m | S_i, \mathbf{z}_{im})$$

We use "Loss of Independence" (LoI) to check this assumption (Huang, 2011).

We introduce it when covariates $\mathbf{z}_i$ are not incorporated in the conditional distribution, more details when covariates $\mathbf{z}_i$ are incorporated can refer to Huang, 2011.

data

Fit LCR model

Collect features

Grid-search on all strategies

Train the classification model

How to build the classification model

$X^2_{LCA}.p$	$D^2_{LCA}.p$	$X^2_{LCR}.p$	$D^2_{LCR}.p$	$G^2.p$	$X^2_{class}.p$	$D^2_{class}.p$	$\overline{X^2_{cp}.p}$	$\overline{D^2_{cp}.p}$	LoI	I_{LCA}	I_{LCR}

Check the conditional independence assumption :

To introduce it, the response vector $(Y_{i1}, \dots, Y_{iM})^T$ is represented as a vector with elements being the indicators of each category :

$$\begin{aligned}\dot{\mathbf{Y}}_i &= \left(\dot{\mathbf{Y}}_{i1}, \dot{\mathbf{Y}}_{i2}, \dots, \dot{\mathbf{Y}}_{iM}\right) \\ &= \left(Y_{i11}, \dots, Y_{i1(K_1-1)}, Y_{i21}, \dots, Y_{i2(K_2-1)}, \dots, Y_{iM1}, \dots, Y_{iM(K_M-1)}\right)\end{aligned}$$

with $Y_{imk} = \mathrm{I}(Y_{im} = k);\ m = 1, \dots, M; k = 1, \dots, (K_m - 1)$. Then

$$\mathrm{Cov}\left(\dot{\mathbf{Y}}_i\right) = \{\mathrm{Cov}(Y_{imk}, Y_{its})\} = \begin{bmatrix} \mathbf{B}_{11} & \mathbf{B}_{12} & \cdots & \mathbf{B}_{1M} \\ \mathbf{B}_{21} & \mathbf{B}_{22} & \cdots & \mathbf{B}_{1M} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{B}_{M1} & \mathbf{B}_{M2} & \cdots & \mathbf{B}_{MM} \end{bmatrix}$$

where $\mathbf{B}_{mt} = \mathrm{Cov}\left(\dot{\mathbf{Y}}_{im}, \dot{\mathbf{Y}}_{it}\right)$ is a $(K_m - 1) \times (K_t - 1)$ block matrix

data

Fit LCR model

Collect features

Grid-search on all strategies

Train the classification model

How to build the classification model

$X^2_{LCA}.p$	$D^2_{LCA}.p$	$X^2_{LCR}.p$	$D^2_{LCR}.p$	$G^2.p$	$X^2_{class}.p$	$D^2_{class}.p$	$\overline{X^2_{cp}.p}$	$\overline{D^2_{cp}.p}$	LoI	I_{LCA}	I_{LCR}

Check the conditional independence assumption :

$$\mathrm{Cov}\left(\dot{\mathbf{Y}}_i\right) = \{\mathrm{Cov}(Y_{imk}, Y_{its})\} = \begin{bmatrix} \mathbf{B}_{11} & \mathbf{B}_{12} & \cdots & \mathbf{B}_{1M} \\ \mathbf{B}_{21} & \mathbf{B}_{22} & \cdots & \mathbf{B}_{1M} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{B}_{M1} & \mathbf{B}_{M2} & \cdots & \mathbf{B}_{MM} \end{bmatrix}$$

Various components of the above covariance matrix are :

$$\mathrm{Cov}(Y_{imk}, Y_{its}) = \begin{cases} \Pr(Y_{imk}=1) - \Pr(Y_{imk}=1)\Pr(Y_{its}=1) & \text{if } m = t \text{ and } k = s, \\ -\Pr(Y_{imk}=1)\Pr(Y_{its}=1) & \text{if } m = t \text{ and } k \neq s, \\ \Pr(Y_{imk}=1, Y_{its}=1) - \Pr(Y_{imk}=1)\Pr(Y_{its}=1) & \text{if } m \neq t. \end{cases}$$

The sample covariance matrix of $\mathrm{Cov}\left(\dot{\mathbf{Y}}_i\right)$ can be obtained by replacing the probabilities with their sample averages.

data

Fit LCR model

Collect features

Grid-search on all strategies

Train the classification model

How to build the classification model

$X^2_{LCA}.p$	$D^2_{LCA}.p$	$X^2_{LCR}.p$	$D^2_{LCR}.p$	$G^2.p$	$X^2_{class}.p$	$D^2_{class}.p$	$\overline{X^2_{cp}.p}$	$\overline{D^2_{cp}.p}$	LoI	I_{LCA}	I_{LCR}

Check the conditional independence assumption :

$$\mathrm{Cov}(\dot{\mathbf{Y}}_i) = \{\mathrm{Cov}(Y_{imk}, Y_{its})\} = \begin{bmatrix} \mathbf{B}_{11} & \mathbf{B}_{12} & \cdots & \mathbf{B}_{1M} \\ \mathbf{B}_{21} & \mathbf{B}_{22} & \cdots & \mathbf{B}_{1M} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{B}_{M1} & \mathbf{B}_{M2} & \cdots & \mathbf{B}_{MM} \end{bmatrix}$$

- Let ACov_j be the average of absolute values of entries in off-diagonal blocks of the sample covariance matrix using objects with $C_i = j$.
- ACov_j represents the magnitude of between-indicator covariances for the *j*th class.

$$LoI = \sum_{j=1}^{J} w_j \mathrm{ACov}_j$$

data

Fit LCR model

Collect features

Grid-search on all strategies

Train the classification model

How to build the classification model

$X^2_{LCA}.p$	$D^2_{LCA}.p$	$X^2_{LCR}.p$	$D^2_{LCR}.p$	$G^2.p$	$X^2_{class}.p$	$D^2_{class}.p$	$\overline{X^2_{cp}.p}$	$\overline{D^2_{cp}.p}$	LoI	I_{LCA}	I_{LCR}

Identifiability for LCA/LCR model :

I_{LCA} I_{LCR}

- Identifiability has been a problem of LCA model for a long time.
- Different values of the parameters must generate different probability distributions of the observable variables.

data

Fit LCR model

Collect features

Grid-search on all strategies

Train the classification model

How to build the classification model

$X^2_{LCA}.p$	$D^2_{LCA}.p$	$X^2_{LCR}.p$	$D^2_{LCR}.p$	$G^2.p$	$X^2_{class}.p$	$D^2_{class}.p$	$\overline{X^2_{cp}.p}$	$\overline{D^2_{cp}.p}$	LoI	I_{LCA}	I_{LCR}

Identifiability for LCA model :

The sufficient conditions for the local identifiability of the constrained LCA model are proposed by Huang (2004) :

Let $\boldsymbol{\psi}_j$ be a $\left(\left(\prod_{m=1}^{M} K_m\right) - 1\right) \times 1$ vector with hth element

$$\psi_{hj} = \Pr(\mathbf{Y}_i = \mathbf{y}_h | S_i = j) = \prod_{m=1}^{M} p_{my_{hm}j},$$

where $\mathbf{y}_h = (y_{h1}, \ldots, y_{hM})$ is the hth possible response patterns
ξ is the number of pre-fixed conditional probabilities $p_{mkj} = 0$ or 1. Suppose that

(i) $\left(\prod_{m=1}^{M} K_m\right) - 1 \geq J\left(\sum_{m=1}^{M}(K_m - 1)\right) + J - 1 - \xi;$
(ii) $p_{mkj} > 0$ and $\eta_j > 0$ for all free parameters;
(iii) $\boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_J$ are linearly independent.

Then, the constrained LCA model is locally identifiable.

data

Fit LCR model

Collect features

Grid-search on all strategies

Train the classification model

How to build the classification model

$X^2_{LCA}.p$	$D^2_{LCA}.p$	$X^2_{LCR}.p$	$D^2_{LCR}.p$	$G^2.p$	$X^2_{class}.p$	$D^2_{class}.p$	$\overline{X^2_{cp}.p}$	$\overline{D^2_{cp}.p}$	LoI	I_{LCA}	I_{LCR}

Identifiability for LCA model :

(i) $(\prod_{m=1}^{M} K_m) - 1 \geq J(\sum_{m=1}^{M}(K_m - 1)) + J - 1 - \xi$;

(ii) $p_{mkj} > 0$ and $\eta_j > 0$ for all free parameters;

(iii) $\boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_J$ are linearly independent.

- (i) and (ii) can be easily restricted by programing setting.
- To check (iii), define the matrix $\boldsymbol{\Psi} = [\boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_J]$. Then $\boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_J$ are linearly independent if and only if the eigenvalues of $\boldsymbol{\Psi}^T\boldsymbol{\Psi}$ are all positive.

$$I_{LCA} = (\text{the number of eigenvalues of } \boldsymbol{\Psi}^T\boldsymbol{\Psi} > 0.1)/J$$

data

Fit LCR model

Collect features

Grid-search on all strategies

Train the classification model

How to build the classification model

$X^2_{LCA}.p$	$D^2_{LCA}.p$	$X^2_{LCR}.p$	$D^2_{LCR}.p$	$G^2.p$	$X^2_{class}.p$	$D^2_{class}.p$	$\overline{X^2_{cp}.p}$	$\overline{D^2_{cp}.p}$	LoI	I_{LCA}	I_{LCR}

Identifiability for LCR model :

Let $\boldsymbol{\tau}_j$ be a $\left(\left(\prod_{m=1}^{M} K_m\right) - 1\right) \times 1$ vector with hth element

$$\tau_{hj} = \prod_{m=1}^{M} \left\{ \frac{\exp\left(\gamma_{m y_{hmj}}\right)}{1 + \sum_{k=1}^{K-1} \exp\left(\gamma_{mkj}\right)} \right\}$$

To check $\boldsymbol{\tau}_1, \ldots, \boldsymbol{\tau}_J$ are linearly independent, Define the matrix $\boldsymbol{\Lambda} = \left[\boldsymbol{\tau}_1, \ldots, \boldsymbol{\tau}_J\right]$ and calculate the ratio of eigenvalues of $\boldsymbol{\Lambda}^T\boldsymbol{\Lambda}$ greater than 0.1

$$I_{LCR} = (\text{the number of eigenvalues of } \boldsymbol{\Lambda}^T\boldsymbol{\Lambda} > 0.1)/J$$

data

Fit LCR model

Collect features

Grid-search on all strategies

Train the classification model

How to build the classification model

Hyper-parameters :

$$\Pr(Y_{i1} = y_1, \ldots, Y_{iM} = y_m) = \sum_{j=1}^{J} \left\{ \eta_j(\mathbf{x}_i) \prod_{m=1}^{M} \prod_{k=1}^{K_m} p_{mkj}^{y_{mk}}(\mathbf{z}_{im}) \right\}$$

with $\mathbf{x}_i = (1, x_{i1}, \ldots, x_{ip})^T$

$\mathbf{z}_i = (\mathbf{z}_{i1}, \ldots, \mathbf{z}_{iM})$ with $\mathbf{z}_{im} = (1, z_{im1}, \ldots, z_{imL})^T$

1. **Number of latent class, J**
2. **covariate of latent class prevalence, $\mathbf{x}_i$**
3. **covariate of conditional probabilities of measured response, $\mathbf{z}_{im}$**

data

Fit LCR model

Collect features

Grid-search on all strategies

Train the classification model

How to build the classification model

Strategies for each hyper-parameters :

Given the current number of latent classes J^*, the new J:

1. stay the same (i.e., $J = J^*$)
2. $J = J^* - 1$ if $J^* \geq \max\{(K_1 + 1), \ldots, (K_M + 1)\}$
3. $J = J^* + 1$ if $J^* \leq \sum_{m=1}^{M}(K_m - 1)$ and $J^* \leq \frac{(\prod_{m=1}^{M} K_m)+\xi}{1+\sum_{m=1}^{M}(K_m-1)} - 1$

For covariates of latent class prevalence, $\mathbf{x}_i$:

1. stay the same
2. remove the covariate with the largest average p-value across all levels of each response if the number of covariates of latent class prevalence ≥ 2

For covariates of conditional probabilities of measured response, $\mathbf{z}_{im}$:

1. stay the same
2. remove the covariate with the largest average p-value across all levels of each response if the number of covariates of conditional probabilities ≥ 2

data

Fit LCR model

Collect features

Grid-search on all strategies

Train the classification model

How to build the classification model

Strategies for each hyper-parameters :

Given the current number of latent classes J^*, the new J:

1. stay the same (i.e., $J = J^*$)
2. $J = J^* - 1$ if $J^* \geq \max\{(K_1 + 1), \ldots, (K_M + 1)\}$
3. $J = J^* + 1$ if $J^* \leq \sum_{m=1}^{M}(K_m - 1)$ and $J^* \leq \frac{(\prod_{m=1}^{M} K_m)+\xi}{1+\sum_{m=1}^{M}(K_m-1)} - 1$

where ξ is the number of pre-fixed conditional probabilities $p_{mkj} = 0$ or 1

- **The bound of J is reasonable if we think of latent variable modeling as a dimension reduction process.**

- **The bound $J \leq \frac{\prod_{m=1}^{M} K_m+\xi}{1+\sum_{m=1}^{M}(K_m-1)} - 1$ states that the number of model's parameters cannot exceed the number of independent pieces of observed information.**

data

Fit LCR model

Collect features

Grid-search on all strategies

Train the classification model

How to build the classification model

Training data generation :

- 12 possible strategies for adjusting hyper-parameters are the output labels.
- Adjust the hyper-parameters according to the output label that can result in the smallest residual sum of squares

$$\sum_{h=1}^{K^{\#}} \left(f_h - \hat{f}_h\right)^2$$

and fit LCR model again.

- Repeat above process until the label representing all three hyper-parameters remaining unchanged is selected.
- Note that the training data is imbalanced if the procedure is followed.

How to build the classification model

3 types of hyper-parameter selection approaches :

1. LCR with covariate effects on conditional probabilities only
2. LCR with covariate effects on latent prevalence only
3. LCR with covariate effects on both conditional probabilities and latent prevalence

5 Adopted Machine Learning Classifier :

1. Logistic regression
2. Random forest
3. Support vector machine (SVM)
4. Adaptive boosting (AdaBoost)
5. Extreme gradient boosting (XGBoost)

How to build the classification model

Model selection :

- Since the sizes of training data are small, we use 5-fold cross-validation to evaluate model performance.
- We choose the one with the highest validation accuracy in average as our model.

How to build the classification model

covariate effects on conditional probabilities only		
model	training acc. (s.d.)	validation acc. (s.d.)
logistic regression	0.724 (0.097)	0.147 (0.086)
random forest	0.787 (0.058)	0.321 (0.114)
SVM	0.388 (0.022)	0.339 (0.080)
AdaBoost	0.616 (0.039)	0.235 (0.044)
XGBoost	1 (0)	0.324 (0.116)
covariate effects on latent prevalence only		
model	training acc. (s.d.)	validation acc. (s.d.)
logistic regression	0.801 (0.054)	0.265 (0.144)
random forest	0.776 (0.062)	0.467 (0.126)
SVM	0.385 (0.010)	0.388 (0.041)
AdaBoost	0.564 (0.040)	0.442 (0.096)
XGBoost	1 (0)	0.293 (0.223)
covariate effects on both conditional probabilities and latent prevalence		
model	training acc. (s.d.)	validation acc. (s.d.)
logistic regression	0.939 (0.062)	0.120 (0.072)
random forest	0.691 (0.056)	0.341 (0.111)
SVM	0.455 (0.064)	0.333 (0.056)
AdaBoost	0.500 (0.050)	0.241 (0.158)
XGBoost	1 (0)	0.250 (0.147)

data

Fit LCR model

Collect features

Grid-search on all strategies

Train the classification model

研究成果

Result

Testing data

The data ($N = 1641$) is about Vision-related disability assessed via. the Activities of Daily Vision Scale (ADVS)

Measured responses : 5 tasks comprising the “far vision” subscale of the ADVS

1. reading street signs at night (3 levels)
2. reading street signs in daylight (2 levels)
3. walking down steps during daylight (2 levels)
4. walking down steps in dim light (2 levels)
5. watching TV (2 levels)

Covariate effects on latent prevalence :

1. number of reported comorbid diseases
2. visual acuity (視力)
3. contrast sensitivity (對比敏感度)
4. glare sensitivity (眩光敏感度)
5. stereoacuity (立體視)
6. central visual field (中央視野)

Covariate effects on conditional probabilities :

1. age
2. cognitive status assessed with the MMSE score
3. years of education
4. gender
5. race
6. GHQ depression subscale score

Model performance on testing data

- The difference between
    1. residual sum of squares based on the predicted label
    2. residual sum of squares based on the label obtained by using grid-search

  can be used to evaluate our hyper-parameters selection approach

- Efficiency of our model on the testing data is defined as :

$$eff. = \frac{\epsilon_0 - \hat{\epsilon}_\#}{\epsilon_0 - \epsilon_\#}$$

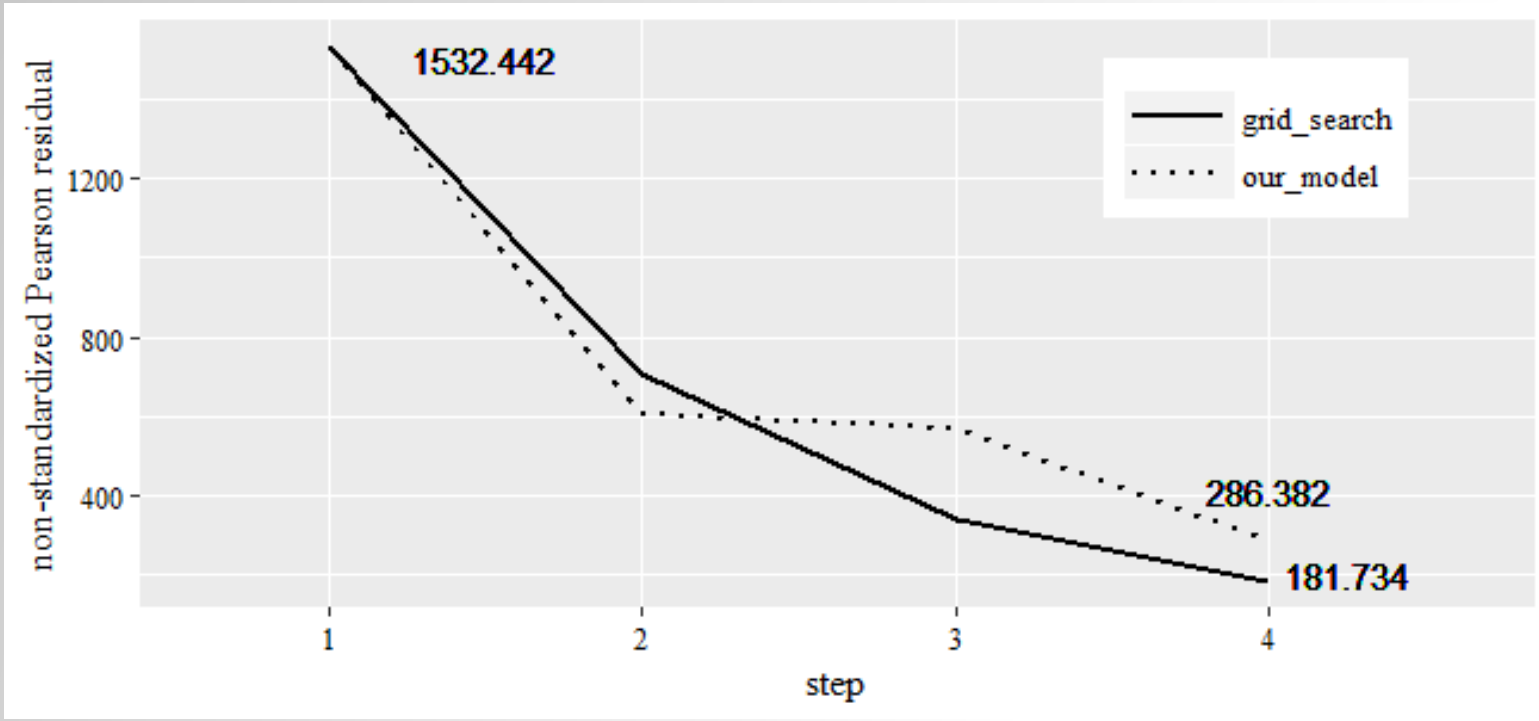
ϵ_0 is the baseline residual sum of squares of the LCR model without changing any hyper-parameters

$\epsilon_\#$ is the residual sum of squares after hyper-parameter selection using grid-search

$\hat{\epsilon}_\#$ is the residual sum of squares obtained by using our approach

Model performance on testing data

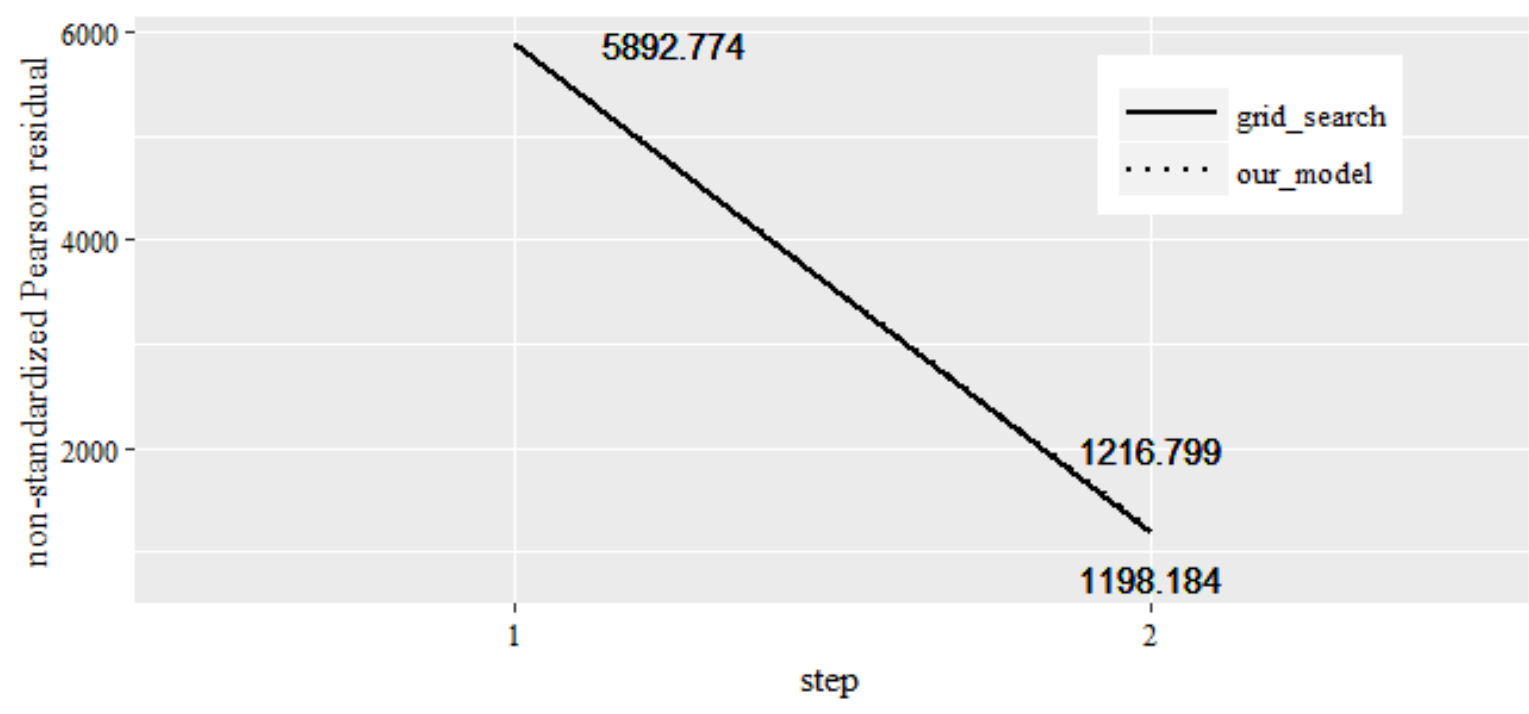
LCR with covariate effects on conditional probabilities only ($eff. = 92.3\%$)



The difference in residual between two approaches is due to the different hyper-parameters selection strategy used.

Model performance on testing data

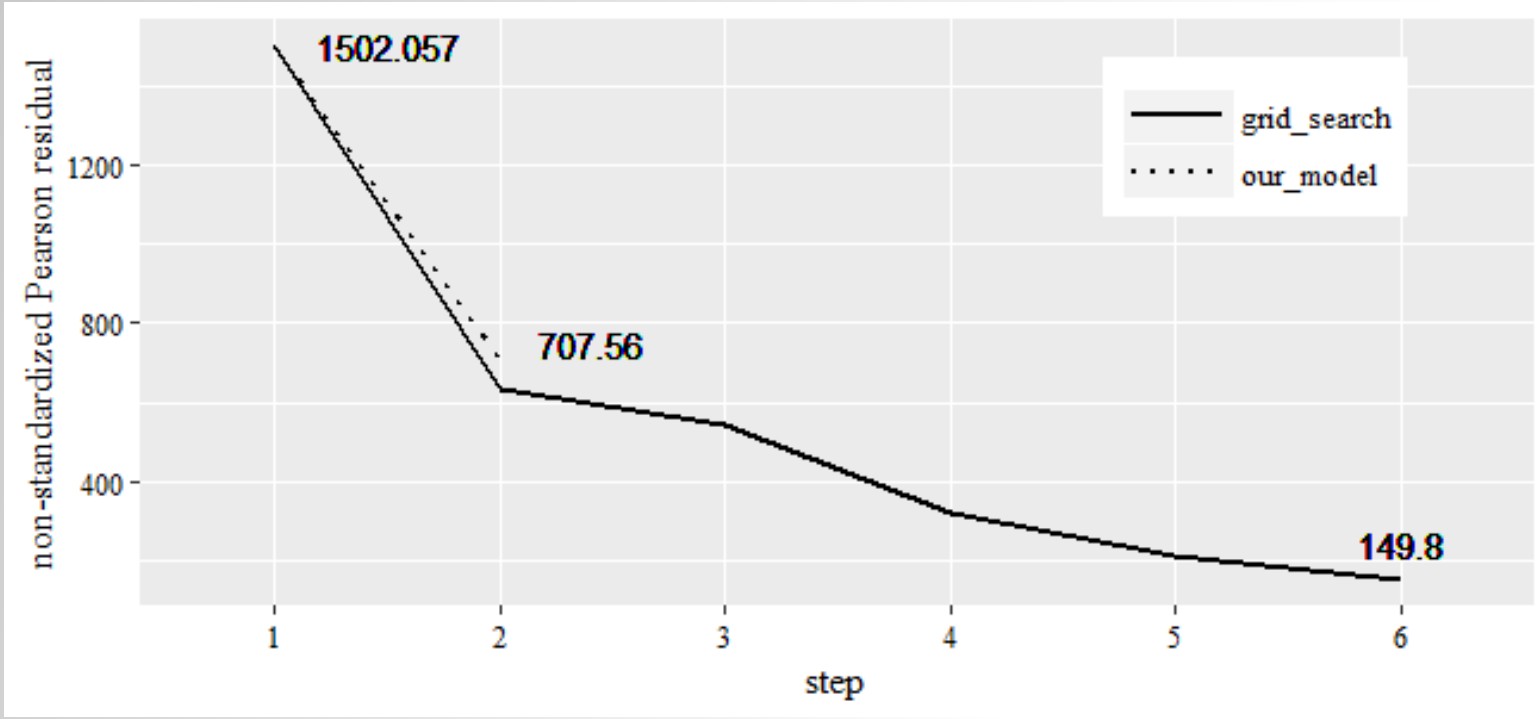
LCR with covariate effects on latent prevalence only ($eff. = 99.6\%$)



The difference in residual between two approaches is due to the random factor in the process of estimating model parameters.

Model performance on testing data

LCR with covariate effects on both conditional probabilities and latent prevalence ($eff. = 58.8\%$)



The difference in residual between two approaches is due to the different hyper-parameters selection strategy used.

結論

Summary

Summary

- Compared with grid-search, our approach cost nearly 5 to 10 times less time to select optimal hyper-parameters of LCR model.

- Our approach’s performance is not as good as expected. It tends to stop the optimization process early since, in training data, the label corresponding to all hyper-parameters remaining unchanged accounts for approximately 35% of the total output labels.

- We expect this dilemma will be overcome when we have more training data.