British Journal of Mathematical and Statistical Psychology

the british psychological society

A new person-fit method based on machine learning in CDM in education

Zhemin Zhu[1*], David Arthur[2] and Hua-Hua Chang[2]

[1]Beihua University, Jilin, China

[2]Purdue University, West Lafayette, Indiana, USA

Reporter: Yingshi Huang

Introduction

- Cognitive diagnostic computerized adaptive testing (CD-CAT)
  - support classroom teaching
- Problems (in-class assessment vs regular exam):
  - shorter test length
  - more aberrant responses
  - less information contained in response time

Person-fit method for in-class exercises?

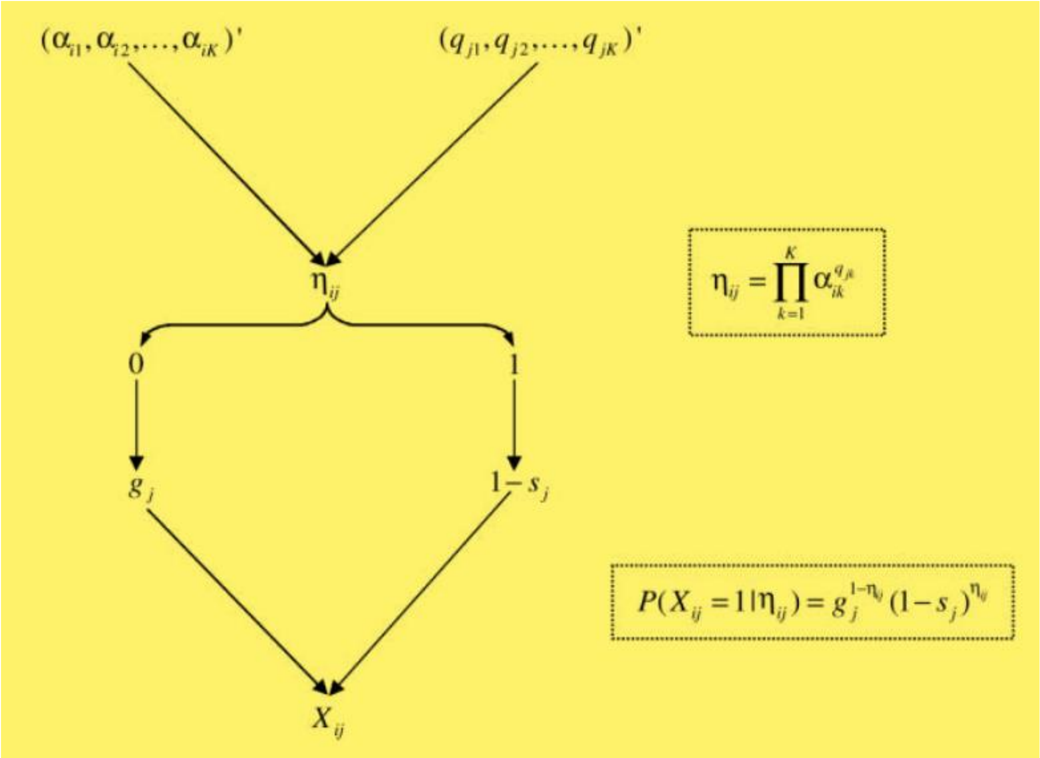
- Purpose:
  - propose a new person-fit method (based on machine learning)

Cognitive diagnostic modelling

- The deterministic input, noisy 'and' gate (DINA) model
  - use MCMC to estimate parameters

$$P\left(X_{ij}=1|\alpha_i\right) = \left(1-s_j\right)^{\eta_{ij}} g_j^{1-\eta_{ij}}$$

$$\eta_{ij} = \prod_{k=1}^{K} \alpha_{ik}^{q_{jk}}$$



[From de la Torre, 2009 *JEBS*]

The estimation will be interfered by

Aberrant Responses

“observed responses ≠ expected ones”

Person-fit statistics

- Nonparametric methods
- Parametric methods
  - build criteria based on:

1. true attribute pattern
2. ability distribution / cut-off point

→ hard to obtain & unstable

[From *google.com*]

- People are complex
  - difficult to classify responses with a fixed model

Let’s turn to **Machine Learning** for help!

Machine learning

- Reinforcement learning
- Unsupervised learning
- Supervised learning

- The person-fit problem: **classification** **& supervised learning problem**

− normal responses

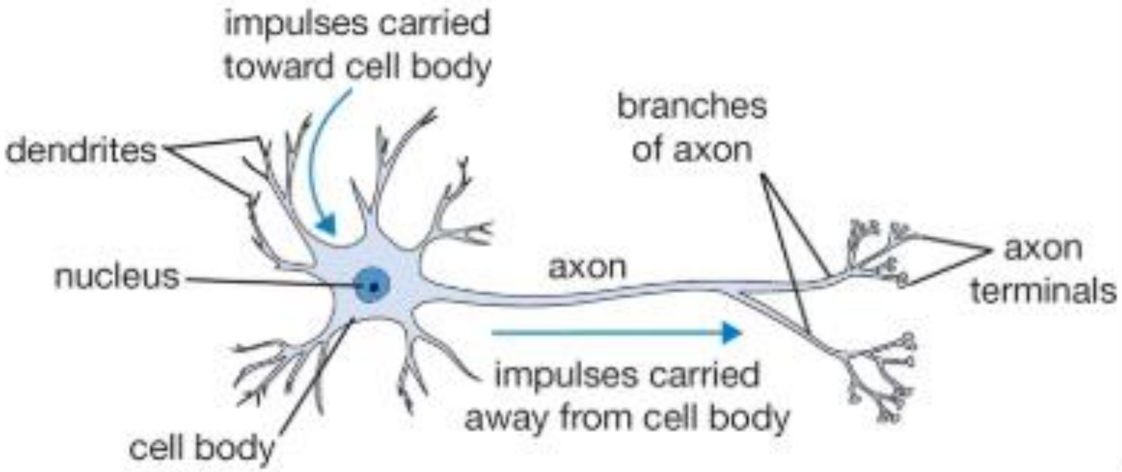
− aberrant responses

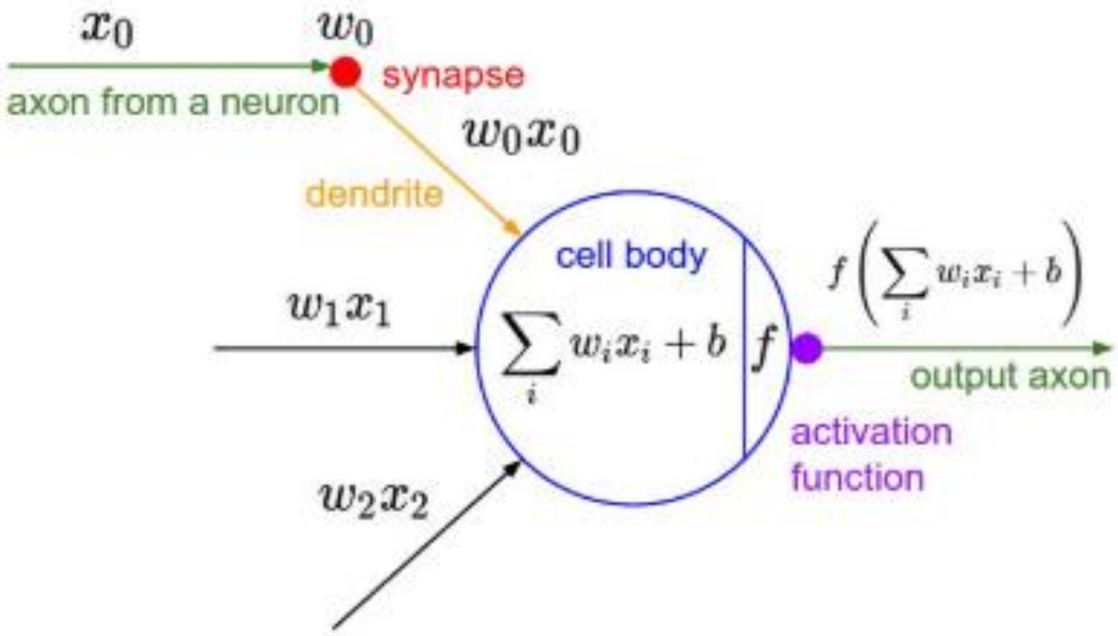
➡ be generated through simulation in actual study

➡ distinguish between *simulated aberrant responses* and *simulated normal responses*

Machine learning

- Supervised learning: neural network
  - neuron in the brain & its mathematical model

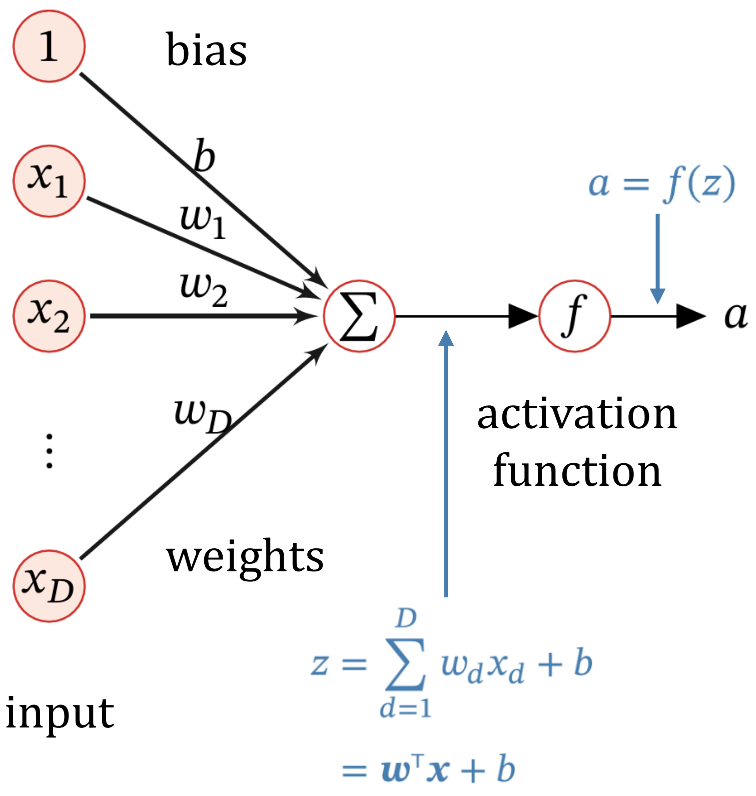




[From *towardsdatascience.com*]

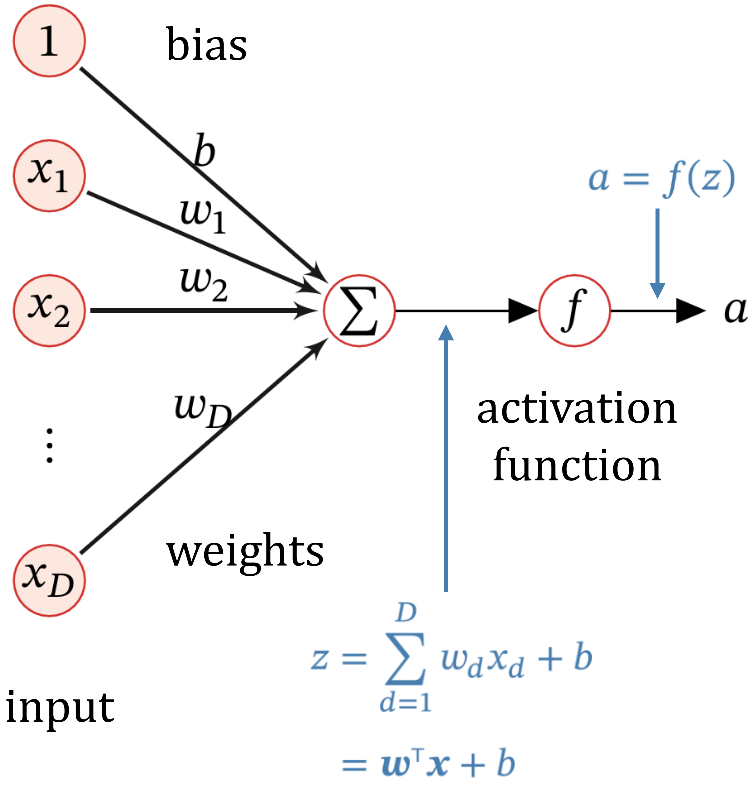
Machine learning

- Supervised learning: neural network



Machine learning

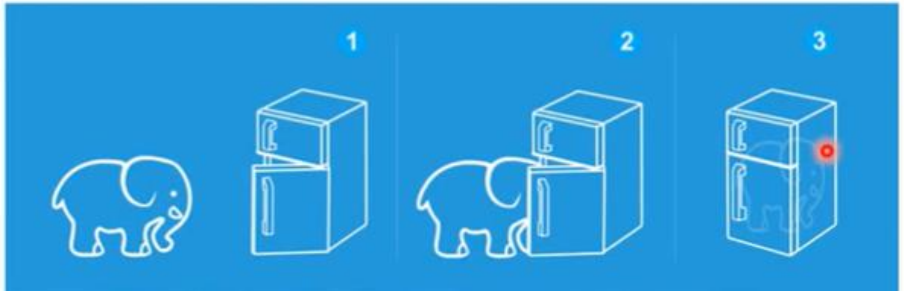
- Supervised learning: neural network



Machine Learning is so simple ......

Step 1: define a set of function → Step 2: goodness of function → Step 3: pick the best function

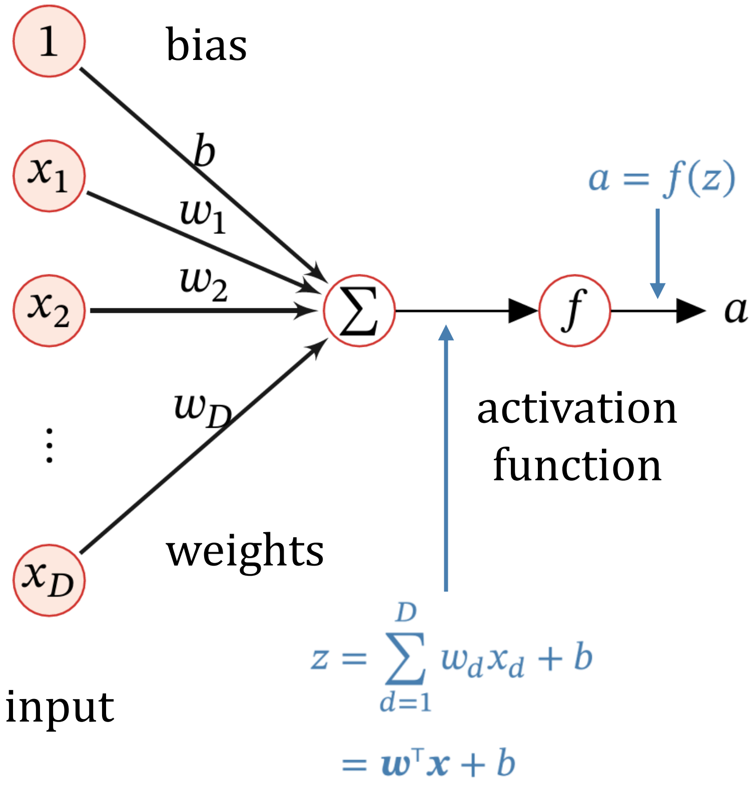
就好像把大象放進冰箱 ......



[From Hung-yi Lee]

Machine learning

- Supervised learning: neural network



- Step 1: sigmoid (or softmax for multi-class)

$$p(y=1|\mathbf{x}) = \sigma(\mathbf{w}^\mathrm{T}\mathbf{x}) \triangleq \frac{1}{1+\exp(-\mathbf{w}^\mathrm{T}\mathbf{x})}$$

- Step 2: cross-entropy

$$\mathcal{R}(W,\mathbf{b}) = -\frac{1}{N}\sum_{n=1}^{N}\left(y^{(n)}\log\hat{y}^{(n)} + (1-y^{(n)})\log(1-\hat{y}^{(n)})\right)$$

- Step 3: gradient descent

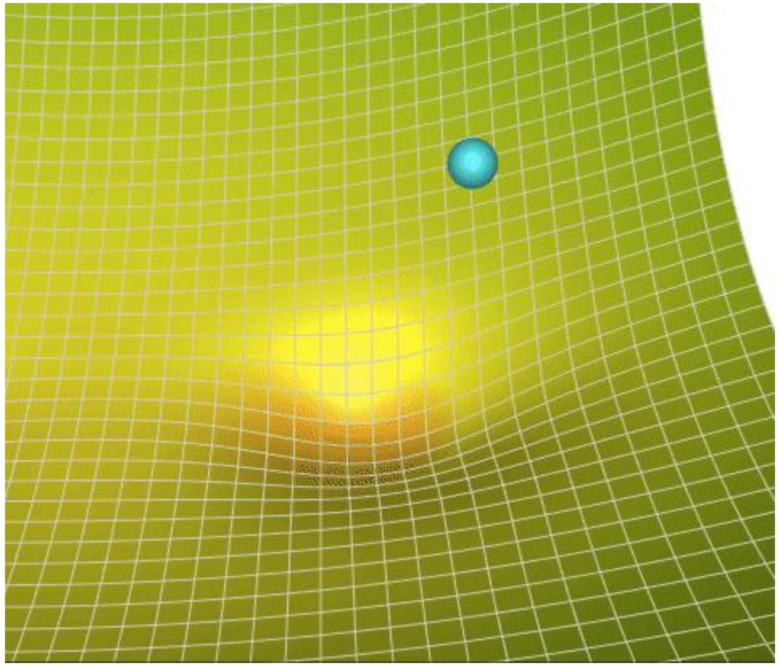
$$W^{(l)} \leftarrow W^{(l)} - \alpha\frac{\partial\mathcal{R}(W,\mathbf{b})}{\partial W^{(l)}}$$

$$\mathbf{b}^{(l)} \leftarrow \mathbf{b}^{(l)} - \alpha\frac{\partial\mathcal{R}(W,\mathbf{b})}{\partial\mathbf{b}^{(l)}}$$

Xipeng Qiu, *Neural Networks and Deep Learning*

Machine learning

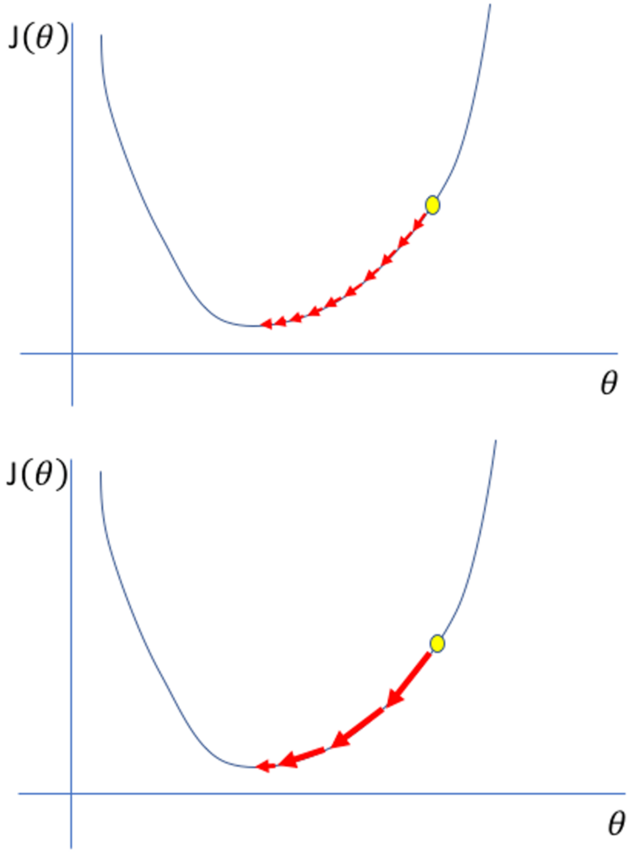
- Supervised learning: neural network
  - gradient descent



[From Lili Jiang]

$$\theta_{t+1} = \theta_t - \alpha \frac{\partial \mathcal{R}_{\mathcal{D}}(\theta)}{\partial \theta}$$

adjust the learning rate



What else can we do but adjust the learning rate?

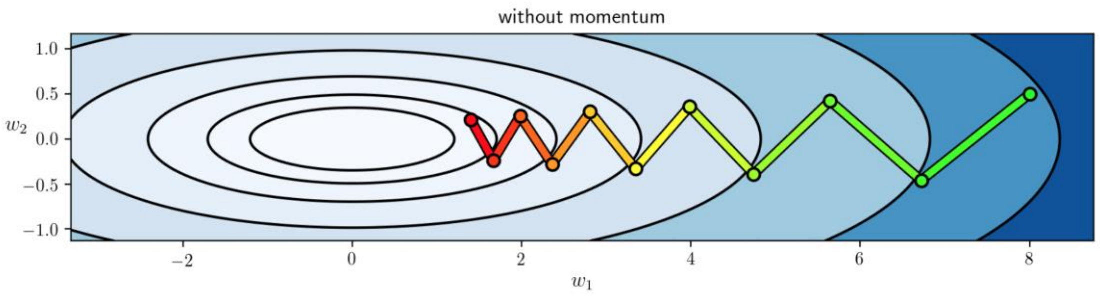
Machine learning

- Supervised learning: neural network
  - modify gradient estimation: momentum method

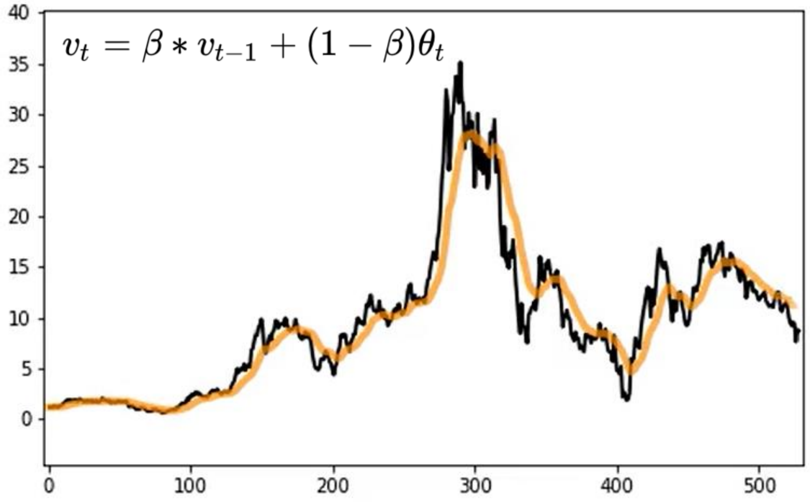
the root cause:
the oscillating nature of the (negative) gradient directions

compute the exponential moving average

create the smoothed descent directions



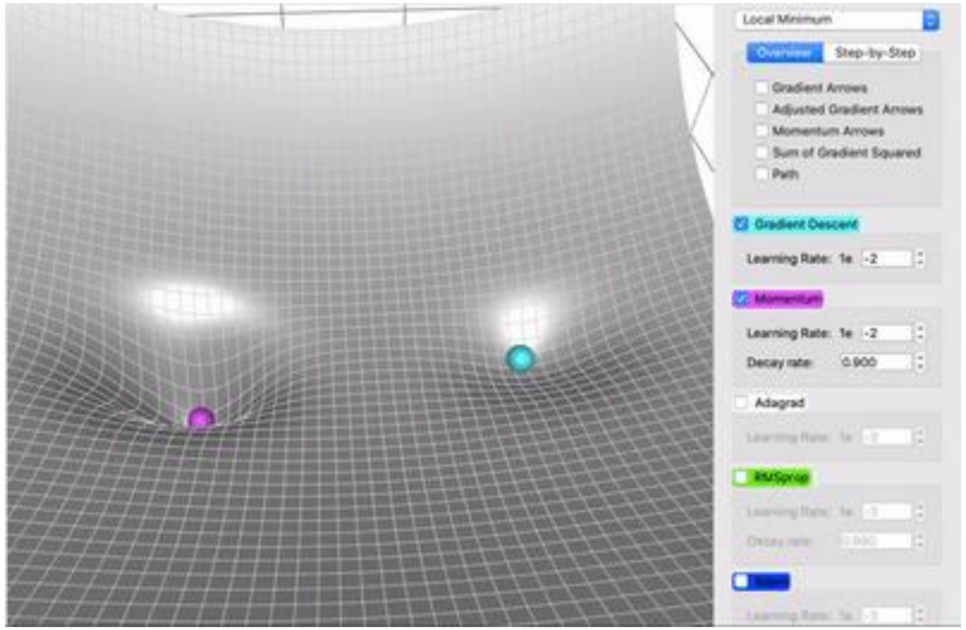
smoothing technique for time series:
exponential average



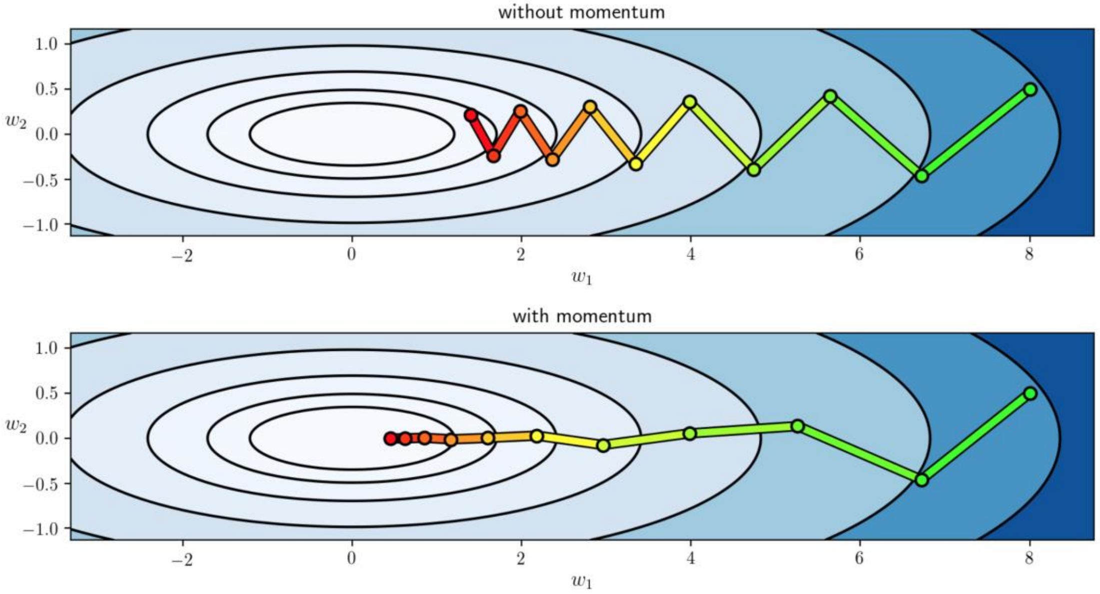
https://jermwatt.github.io/machine_learning_refined/notes/3_First_order_methods/3_8_Momentum.html

Machine learning

- Supervised learning: neural network
  - modify gradient estimation: momentum method



[From Lili Jiang]



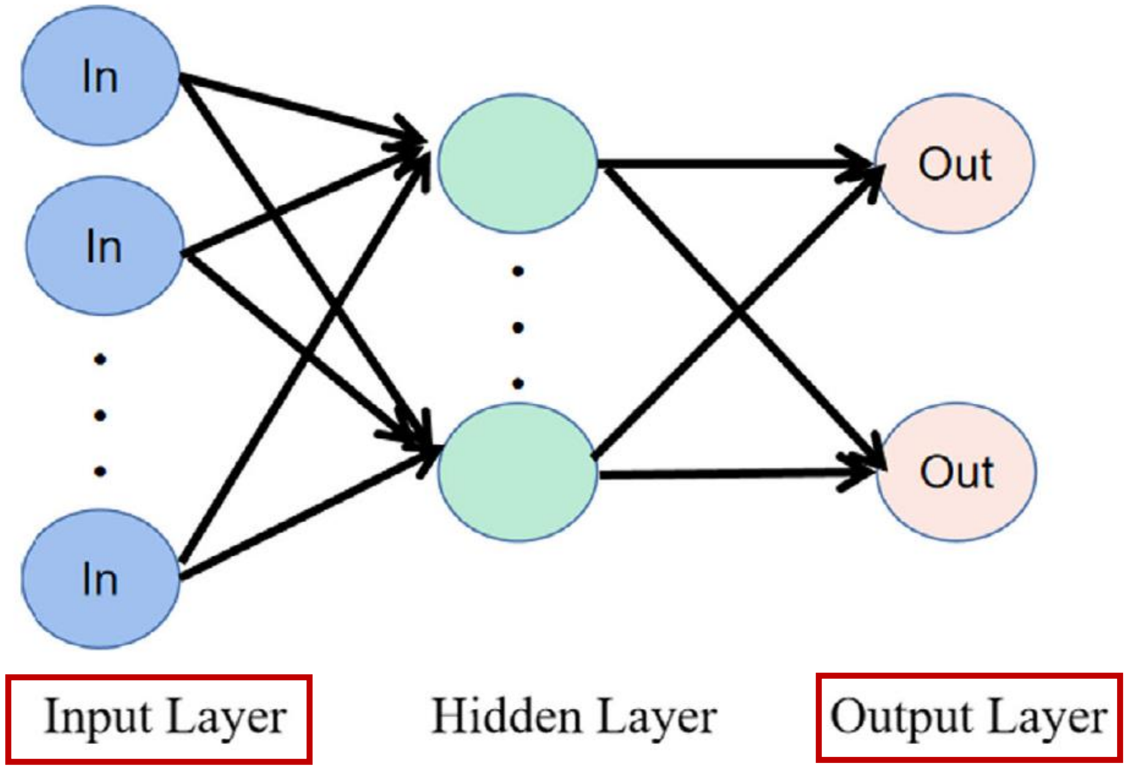
$$\Delta\theta_t = -\alpha \mathbf{g}_t$$

$$\Delta\theta_t = \rho\Delta\theta_{t-1} - \alpha\mathbf{g}_t = -\alpha \sum_{\tau=1}^{t} \rho^{t-\tau}\mathbf{g}_\tau$$

https://jermwatt.github.io/machine_learning_refined/notes/3_First_order_methods/3_8_Momentum.html

Machine learning

- Supervised learning: neural network
  - for aberrant behaviour detection: we just need to design specific input & output

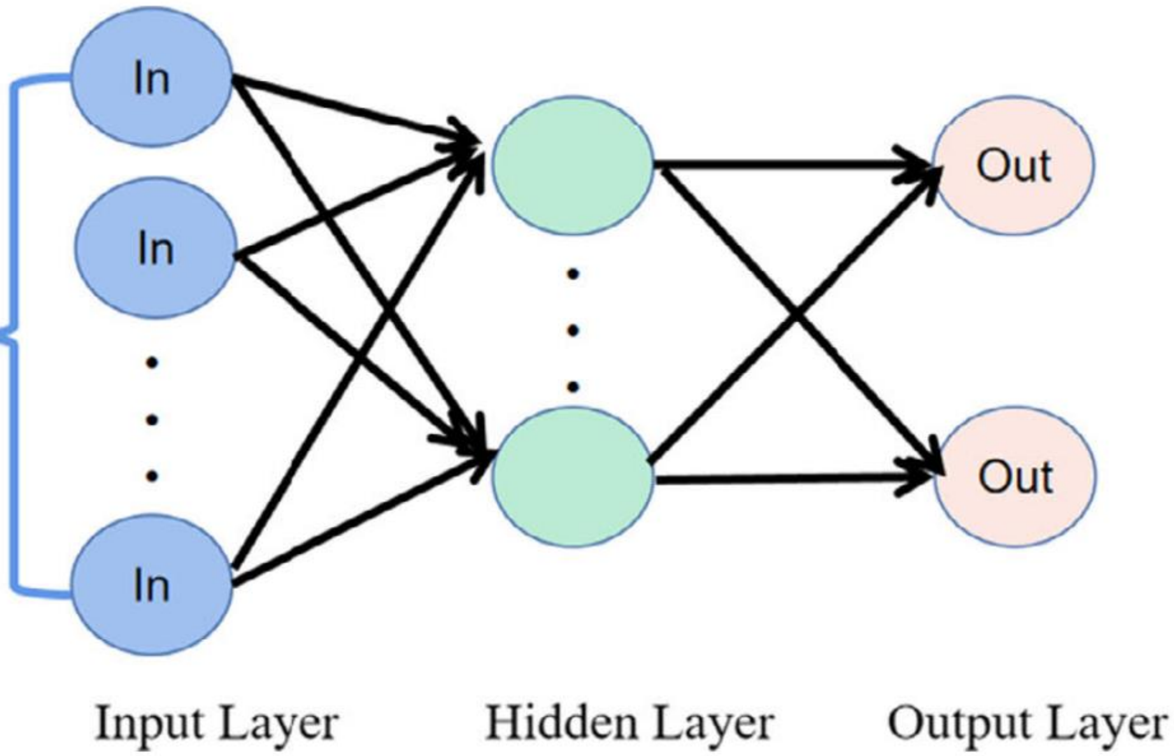


Machine learning

- Supervised learning: neural network
  - for aberrant behaviour detection: **input**

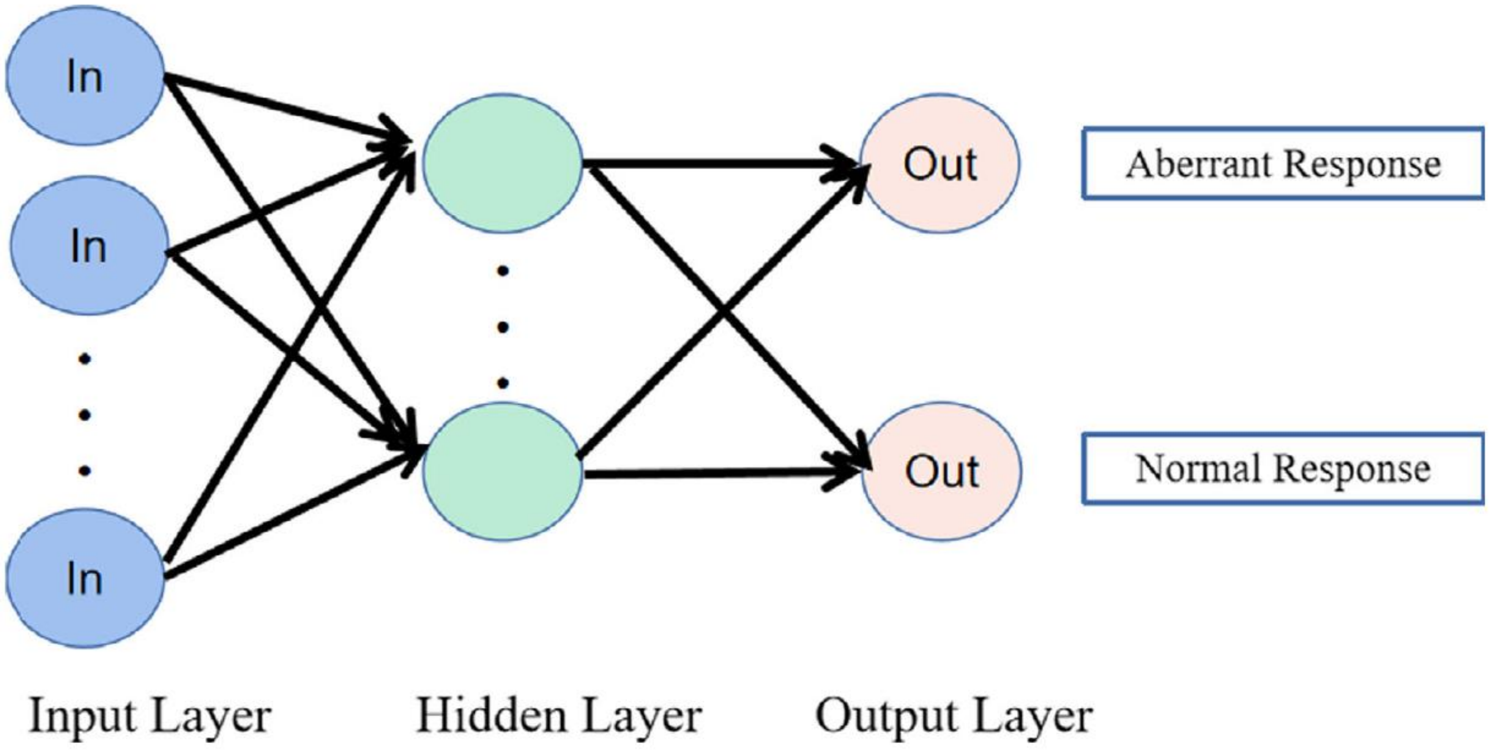
A student's *i* responses (either aberrant or normal) to the *i* items and *k* attributes (mastered or not mastered)

$i + k$ neurons



Machine learning

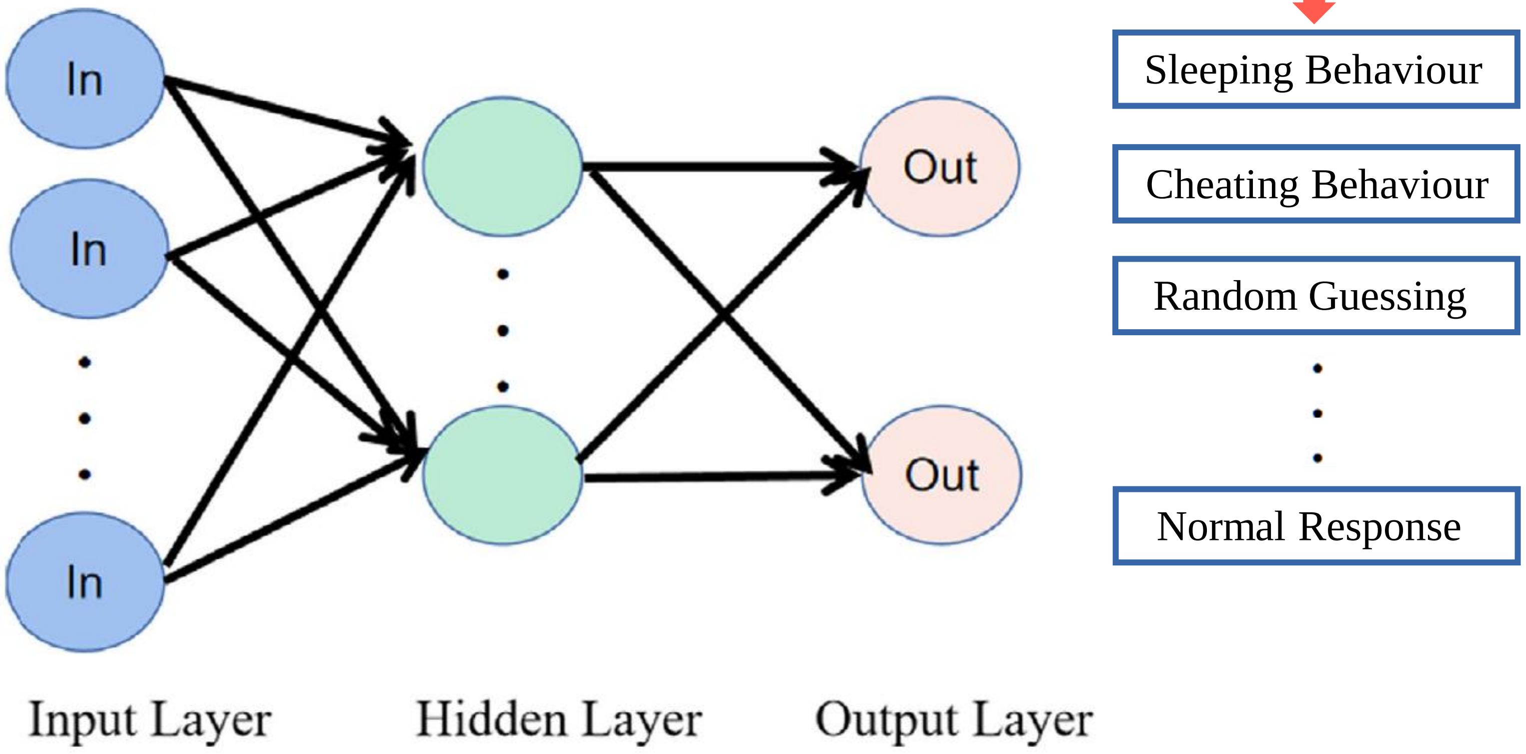
- Supervised learning: neural network
  - for aberrant behaviour detection: **output**



Machine learning

- Supervised learning: neural network
  - for aberrant behaviour detection: **output**

account for the different types of aberrant behaviours



Machine learning

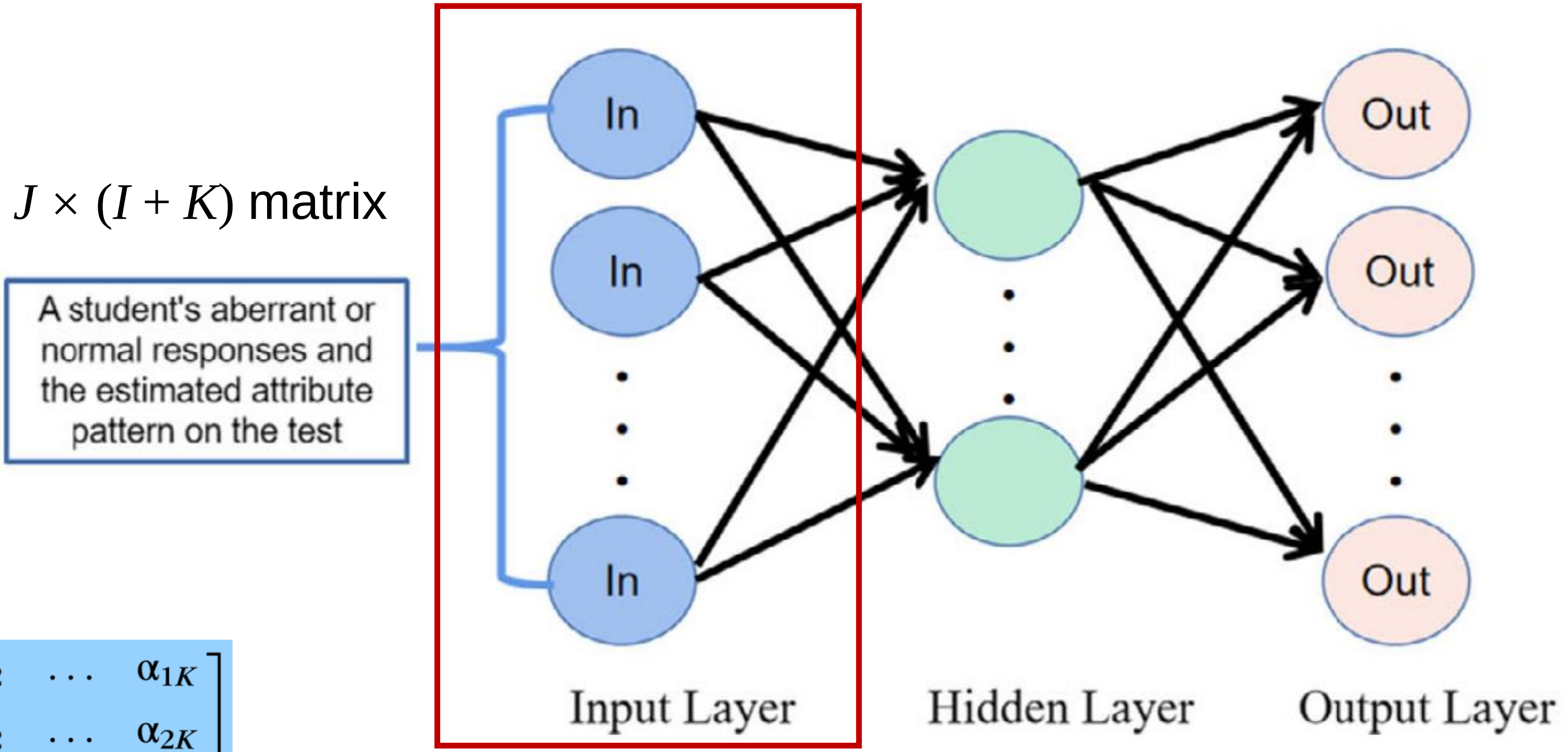
- Supervised learning: neural network
  - two potential advantages:

1. use the same model to determine which kind of aberrant behaviour is being manifested
2. can identify the real attribute patterns

➡ **redesign the output layer** of the neural network

Machine learning

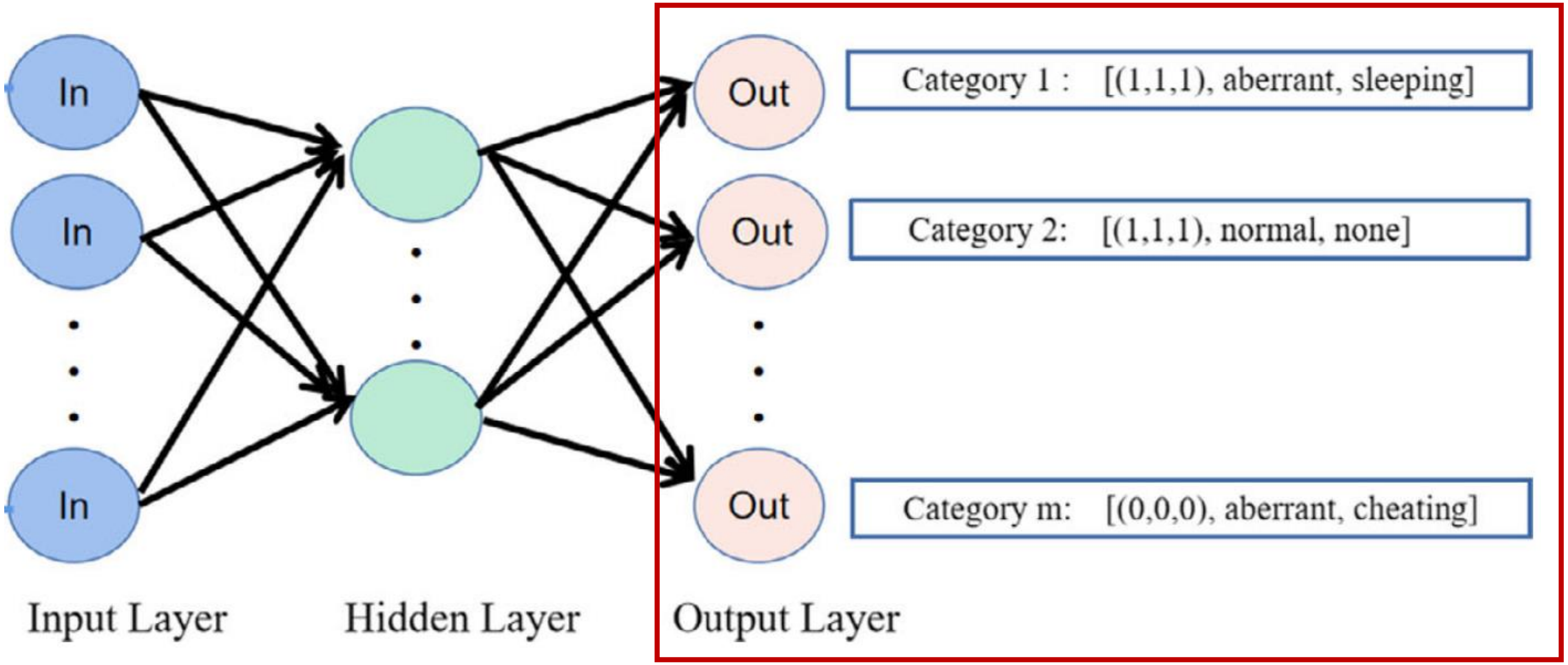
- Machine learning person-fit (MLP-F): **input**



$$
\begin{array}{cc}
 & \begin{matrix} \phantom{u_{11}} & \phantom{u_{12}} & & \text{item } i & \phantom{\alpha_{11}} & \phantom{\alpha_{12}} & & \phantom{\alpha_{1K}} \end{matrix} \\
 & \begin{bmatrix}
u_{11} & u_{12} & \cdots & u_{1I} & \alpha_{11} & \alpha_{12} & \cdots & \alpha_{1K} \\
u_{21} & u_{22} & \cdots & u_{2I} & \alpha_{21} & \alpha_{22} & \cdots & \alpha_{2K} \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
u_{J1} & u_{J2} & \cdots & u_{JI} & \alpha_{J1} & \alpha_{J2} & \cdots & \alpha_{JK}
\end{bmatrix} \text{student } j
\end{array}
$$

Machine learning

- Machine learning person-fit (MLP-F): **output**



- ✓ attribute patterns
- ✓ the response types
- ✓ the kind of abnormal behaviour

Simulation studies

- Simulation 1:
  - determine the **hidden layer structure**

- Simulation 2:
  - examine the **Type I errors** as well as the difference between the training and testing accuracy

- Simulation 3:
  - **compare** the new MLP-F method with the traditional method (RCI)

Simulation studies

- Experiment design (with DINA model)
  - the number of **attributes**: 3, 6
  - the **test length**: 10 (only for 3 attributes), 20 and 40
  - the **proportion** of aberrant responses: 10%, 20%
  - the **discrimination power** of test items:

    high $g, s \sim N(.10, .02)$

    low $g, s \sim N(.25, .05)$

  - 500 replications for each combination

Table A1. *Q*-matrices used in the simulations

3 attributes			6 attributes	
10 items	20 items	40 items	20 items	40 items
100	100	100	100000	100000
010	010	010	010000	010000
001	001	001	001000	001000
101	101	101	000100	000100
011	011	011	000010	000010
110	110	110	000001	000001
101	101	101	010011	101110
110	110	110	000011	010000
011	011	011	111100	011010
111	111	111	001011	000010
	100	100	010110	111100
	010	010	111110	100101
	001	001	111110	100001
	101	101	101010	011101
	011	011	001101	100101
	110	110	001010	011011
	101	101	100111	100101
	110	110	011000	000010
	011	011	101001	100100
	111	111	010100	001110
		100		100110
		010		111011
		001		000011
		101		001001
		011		001011
		110		010110
		101		001010
		110		000001
		011		000101
		111		101101
		100		000111
		010		000010
		001		010111
		101		000110
		011		101010
		110		100011
		101		010001
		110		100001
		011		111001
		111		100010

Simulation studies

- Aberrant behaviour
  - sleeping: high-level students & first several question
  - cheating: examinees who do not have all attributes & the end of the exam
  - random guessing: all examinees & every test question

⬇

  - **sleeping:** at most one missing attribute & the first 10 or 20% (1→0)
  - **cheating:** at least one missing attribute & the last 10 or 20% (0→1)
  - **random guessing:** all attributes patterns & correct response probability == 0.25

⬇

differ in attribute patterns & response sequences

Simulation studies

- Trained neural network: an example with three attributes

Normal (8)

(0, 0, 0)
(1, 0, 0)
(0, 1, 0)
(0, 0, 1)
(1, 1, 0)
(1, 0, 1)
(0, 1, 1)
(1, 1, 1)
(?, ?, ?)

Cheating (7)

Sleeping (4)

Random guessing (1)

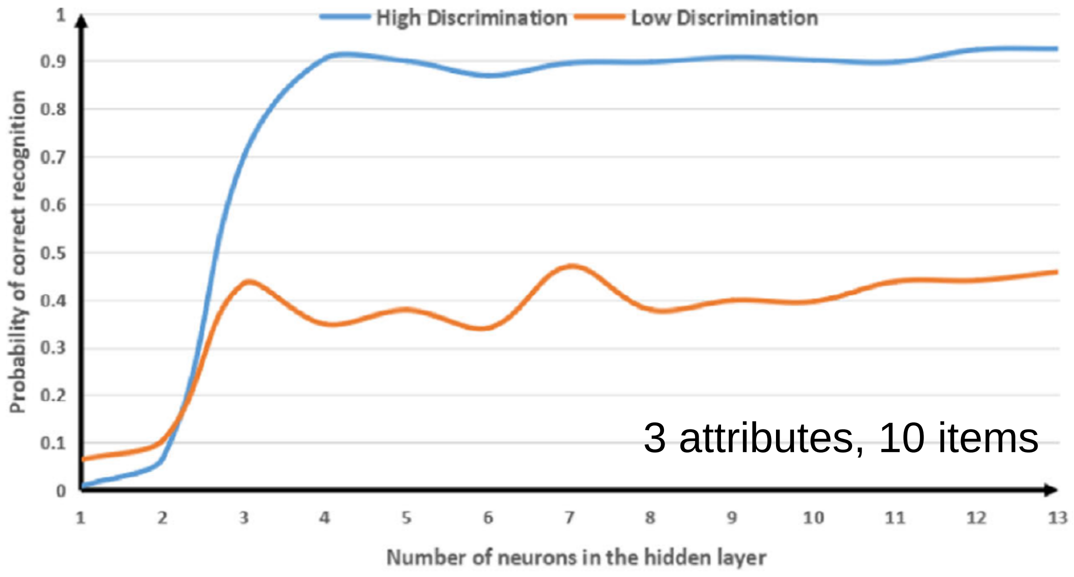
– training & testing:
  50-50 split for each class
– estimated attribute patterns: DINA
– activation function: sigmoid
– proximity: least mean squares
– learning rate: 0.005
– momentum: 0.5

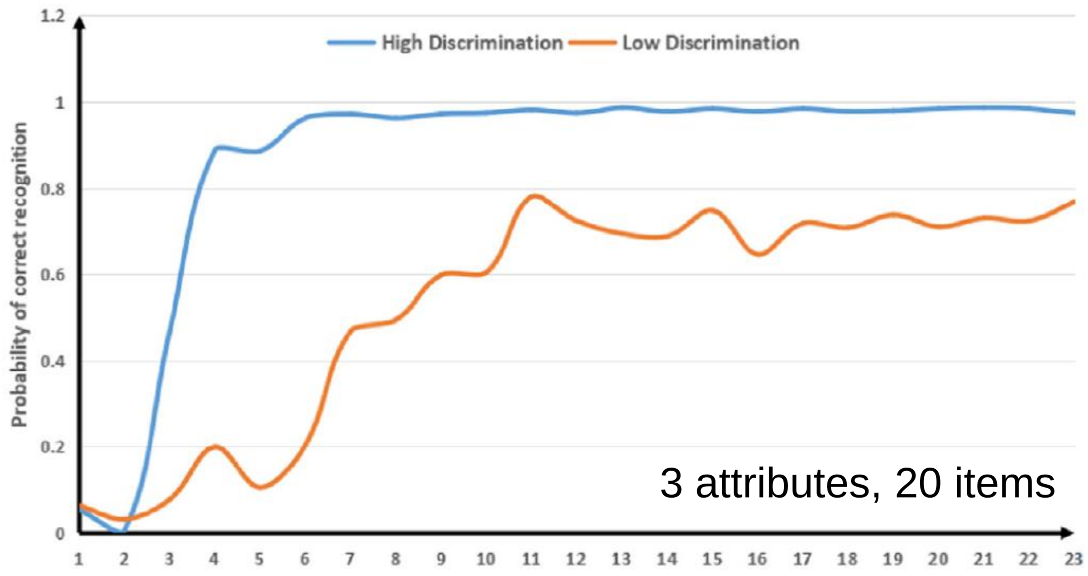
1,000 responses for each class
(20 × 1,000 = 20,000)

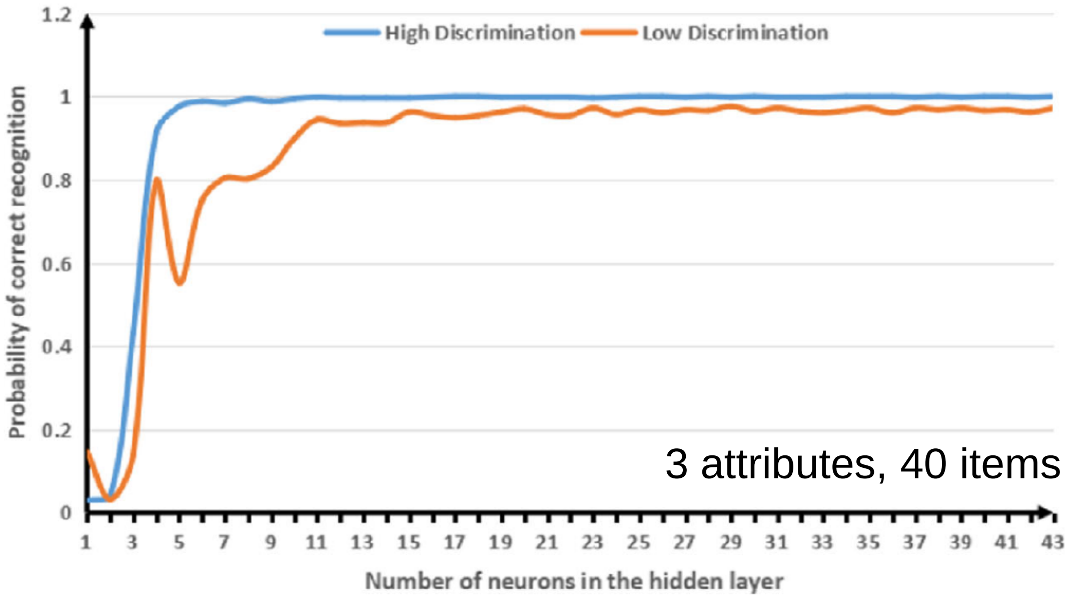
Study 1: The hidden layer structure

- Purpose: find the optimal number of neurons
- Method: go through all of the numbers
  - train different neural networks with neurons ranging from 1 to N_i - 1
- Evaluation: probability of correct recognition
  - normal responses for a random attribute pattern

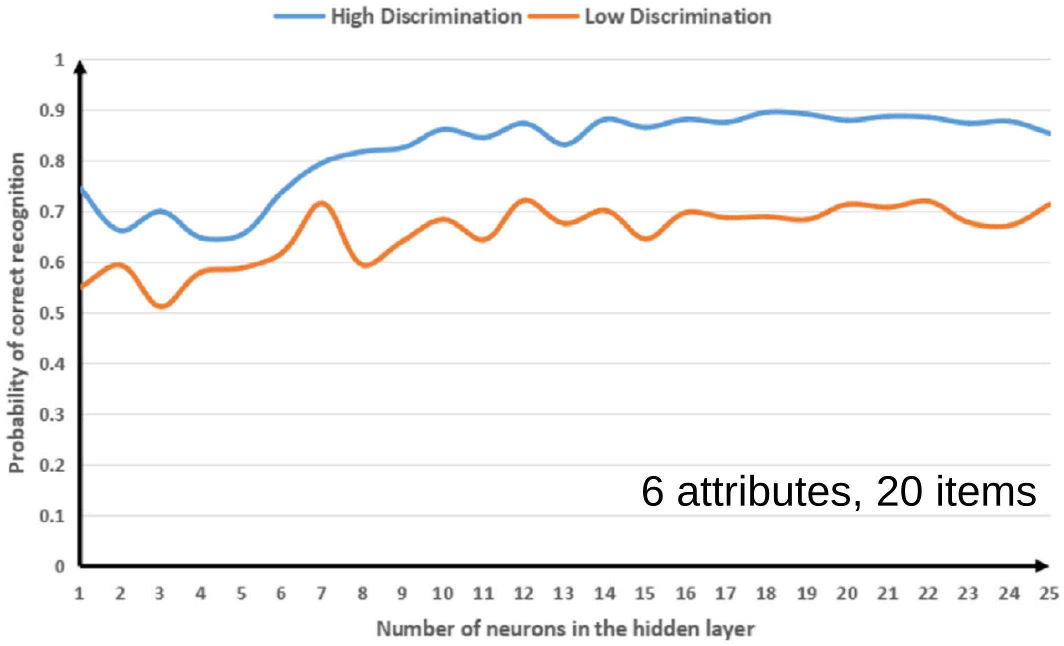
Results: The hidden layer structure



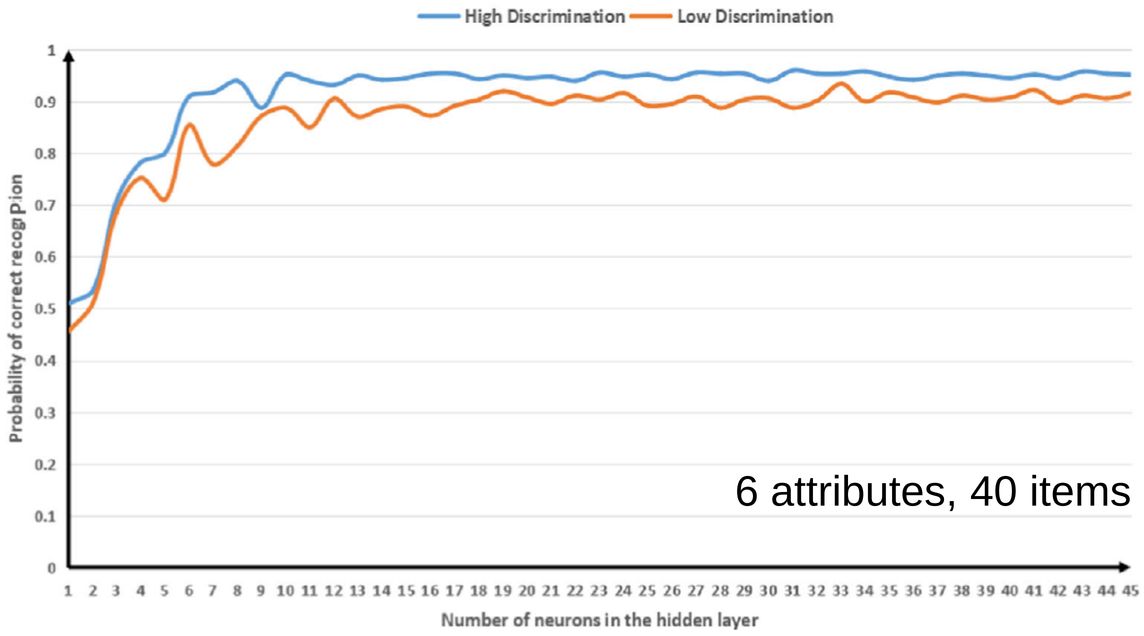




Results: The hidden layer structure



6 attributes, 20 items



6 attributes, 40 items

Results: The hidden layer structure

Table 2. The number of hidden neurons

Attributes	3						6			
Items	10		20		40		20		40	
Discrimination	High	Low	High	Low	High	Low	High	Low	High	Low
Neurons	7	8	7	11	10	16	15	12	12	19

Study 2: Type I errors

- Purpose: the potential problem of false positive and overfitting
- Evaluation: probability of correct recognition & recognition differences
  - the power of MLP-F to recognize normal responses (Type I errors)
  - the difference between the training and testing accuracy (Diff)

Results: Type I errors

Table 3. Summary of Type I errors of MLP-F and Diff from training accuracy

Discrimination		Low		High	
Attributes	Items	Type I errors	Diff	Type I errors	Diff
3	10	28.6	2.6	3.4	0.6
	20	29.0	4.6	0.2	1.6
	40	2.2	2.1	0.1	0.8
6	20	27.6	3.0	11.0	3.4
	40	11.6	4.2	0.1	4.3

Study 3: Recognition rate

- Purpose:
  - explore the power of MLP-F to detect aberrant behaviour and estimate attributes
  - compare MLP-F with control method
- Method:
  - generate 500 aberrant responses under each behaviour for each specific attribute
- Evaluation:
  - MLPO: both the kind of response and the true attribute pattern
  - MLPT: accurately classified as aberrant, regardless of attributes

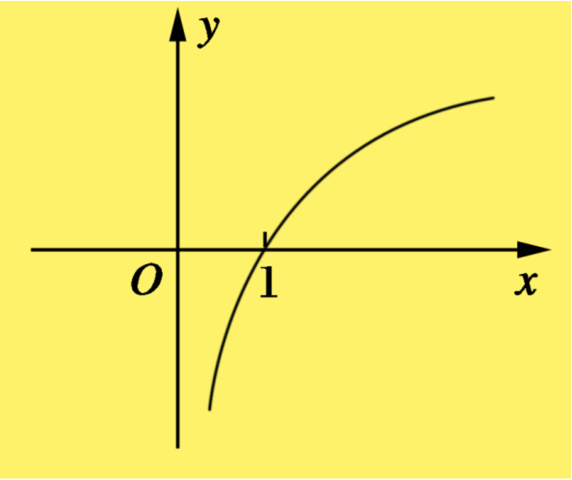
Study 3: Recognition rate

- Control method: the response conformity index (RCI)
  - identify observed response patterns that are incongruent with CDM

➡ observed responses vs Q-matrix expectations

$$\mathrm{RCI}_j = \sum_{i=1}^{I} |\mathrm{RCI}_{ij}| = \sum_{i=1}^{I} \left| \ln\left[-\frac{X_{ij} - P_i(\alpha_j)}{I_i(\alpha_j) - P_i(\alpha_j)} \right]^{X_{ij} + \boxed{I_i(\alpha_j)}} \right|$$

ideal response
= 0 / 1 (master all attributes required)



[From *baidu.com*]

X_{ij}	$I_i(\alpha_j)$	
0	0	$\mathrm{RCI}_i = \ln 1$
1	1	
1	0	
0	1	

Cui & Li, 2015 *APM*

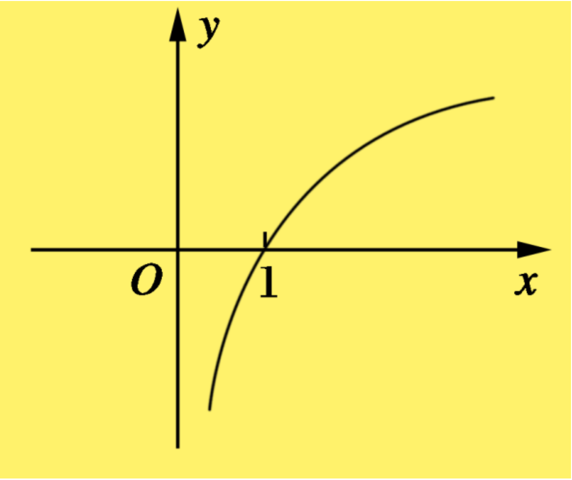
Study 3: Recognition rate

- Control method: the response conformity index (RCI)
  - identify observed response patterns that are incongruent with CDM

➡ observed responses vs Q-matrix expectations

$$\mathrm{RCI}_j = \sum_{i=1}^{I} |\mathrm{RCI}_{ij}| = \sum_{i=1}^{I} \left| \ln\left[- \frac{X_{ij} - P_i(\alpha_j)}{I_i(\alpha_j) - P_i(\alpha_j)} \right]^{X_{ij}+\boxed{I_i(\alpha_j)}} \right|$$

ideal response
= 0 / 1 (master all attributes required)



[From *baidu.com*]

X_{ij}	$I_i(\alpha_j)$
0	0
1	1
1	0
0	1

$$\ln\left[\frac{1 - P_i(\alpha_j)}{P_i(\alpha_j)}\right]$$

- if $P_i(\alpha_j) = 0.5$, RCI_i close to 0

aberrant response

- if $P_i(\alpha_j) \to 0$, RCI_i positive large
- If $P_i(\alpha_j) \to 1$, RCI_i negative large

poor quality of item

Cui & Li, 2015 *APM*

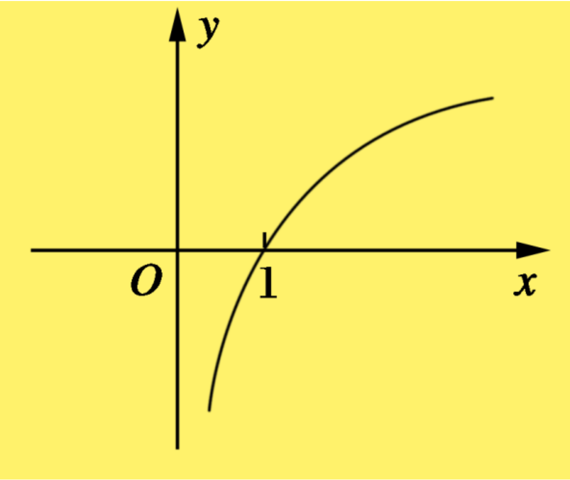
Study 3: Recognition rate

- Control method: the response conformity index (RCI)
  - identify observed response patterns that are incongruent with CDM

➡ observed responses vs Q-matrix expectations

$$\mathrm{RCI}_j = \sum_{i=1}^{I} |\mathrm{RCI}_{ij}| = \sum_{i=1}^{I} \left| \ln\left[-\frac{X_{ij} - P_i(\alpha_j)}{I_i(\alpha_j) - P_i(\alpha_j)} \right]^{X_{ij}+\boxed{I_i(\alpha_j)}} \right|$$

ideal response
= 0 / 1 (master all attributes required)



[From *baidu.com*]

X_{ij}	$I_i(\alpha_j)$	
0	0	
1	1	
1	0	
0	1	$\ln\left[\frac{P_i(\alpha_j)}{1-P_i(\alpha_j)}\right]$

- if $P_i(\alpha_j) = 0.5$, RCI_i close to 0

poor quality of item

- if $P_i(\alpha_j) \to 0$, RCI_i negative large
- if $P_i(\alpha_j) \to 1$, RCI_i positive large

aberrant response

Cui & Li, 2015 *APM*

Study 3: Recognition rate

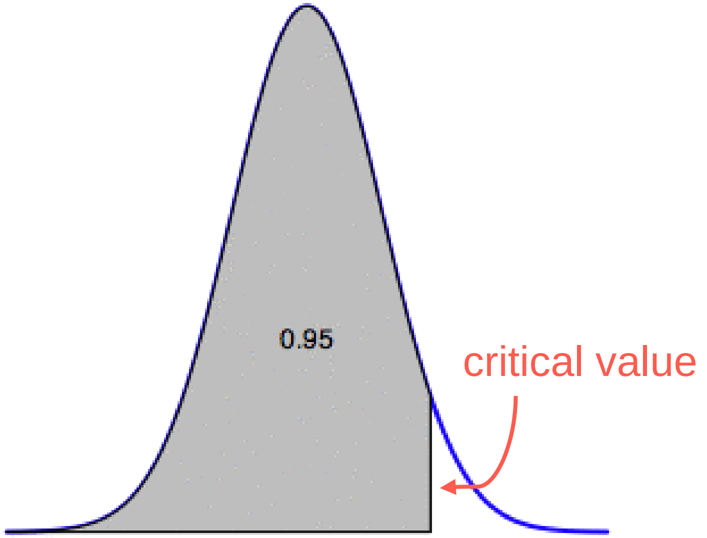
- Control method: the response conformity index (RCI)
  - identify observed response patterns that are incongruent with CDM

observed responses vs Q-matrix expectations

$$\mathrm{RCI}_j = \sum_{i=1}^{I} |\mathrm{RCI}_{ij}| = \sum_{i=1}^{I} \left| \ln \left[-\frac{X_{ij} - P_i(\alpha_j)}{I_i(\alpha_j) - P_i(\alpha_j)} \right]^{X_{ij}+I_i(\alpha_j)} \right|$$

set the **critical values**

1. generate a large number of *normal* responses for each attribute
2. calculate RCIs and order them from low to high
3. choose the 95th percentile value (α = 0.05)



Cui & Li, 2015 *APM*

Results: Recognition rate (sleeping)

Table 4. Summary of RCI and MLP-F in sleeping behaviour simulations with three attributes

Ratio		10%						20%					
Discrimination		Low			High			Low			High		
Attributes	Items	RCI	MLPO	MLPT	RCI	MLPO	MLPT	RCI	MLPO	MLPT	RCI	MLPO	MLPT
(1,1,1)	10	1.2	64.4	99.2	20.3	92.6	100	6.3	75.6	94	28.9	98.4	100
	20	17.4	91.8	99.2	24.1	100	100	44.6	96.8	99.8	84.5	99.4	99.6
	40	21.2	97.8	99.8	54.9	99.6	100	87.3	99.6	100	100	99.6	100
(1,1,0)	10	0.3	40.8	99.0	12.1	90.2	99.8	5.1	46.4	99.6	15.3	91.4	100
	20	17.4	72.6	99.2	29.9	98.2	98.6	1.4	74.8	84.2	56.9	93.8	96.2
	40	13.7	91.4	94.8	40.4	99.4	99.8	16.8	91.6	100	95.4	99.6	100
(0,1,1)	10	2.5	61.8	94.2	6.1	92.4	100	5.8	69.2	98.4	22.2	87.2	100
	20	5.5	74.2	99.8	22.7	97.0	98.4	4.4	72.2	99.4	48.7	97.2	97.6
	40	12.4	96.6	97.2	56.9	99.4	99.6	17.8	93.6	100	96.2	99.6	100
(1,0,1)	10	0.2	34.4	56.2	21.4	97.6	100	1.1	57.0	99.8	38.1	87.8	100
	20	3.7	71.8	99.4	33.5	96.8	97.4	6.3	73.4	88.0	52.6	96.4	98.0
	40	16.3	94.6	96.4	58.5	99.0	99.2	17.3	94.2	100	96.9	99	100

Results: Recognition rate (sleeping)

Table 5. Summary of RCI and MLP-F in sleeping behaviour simulations with six attributes

Ratio		10%						20%					
Discrimination		Low			High			Low			High		
ATT	Items	RCI	MLPO	MLPT	RCI	MLPO	MLPT	RCI	MLPO	MLPT	RCI	MLPO	MLPT
(1,1,1,1,1,1)	20	8.7	9.4	53.8	22.8	99.2	100	10.3	60.2	100	85.7	99.6	100
	40	24.5	88.4	99.8	97.2	97.8	100	56.7	94.2	100	100	100	100
(0,1,1,1,1,1)	20	4.2	37.8	96.2	29.9	94.8	99.8	7.9	53.2	98.4	78.6	97.2	100
	40	17.4	88.2	100	100	98.4	99.4	30.2	92.4	100	100	99.6	100
(1,0,1,1,1,1)	20	0.7	65.2	94.8	30.4	95.2	99.8	5.2	54.8	99.8	67.6	95.4	99.8
	40	21.4	90.4	100	100	98.4	100	31.7	90.6	100	100	99.8	100
(1,1,0,1,1,1)	20	5.6	75.6	99.2	34.3	97.0	99.8	3.5	74.2	96.8	68.1	98.2	99.8
	40	22.1	94.6	99.8	98.7	99.2	99.8	30.4	94.6	99.6	99.2	99.4	100
(1,1,1,0,1,1)	20	0.4	79.0	99.0	43.7	97.2	100	8.8	75.2	99.4	70.5	95.8	99.6
	40	22.7	93.2	99.2	99.9	99.8	100	29.2	96.2	99.8	100	99.6	100
(1,1,1,1,0,1)	20	4.4	84.0	94.0	42.9	95.6	99.8	6.7	78.6	98.2	72.4	98.6	99.2
	40	19.9	96.8	100	100	99.6	100	26.3	97.6	100	97.4	99.8	100
(1,1,1,1,1,0)	20	8.7	96.6	99.8	49.1	99.4	100	14.4	95.6	100	81.0	98.4	99.4
	40	20.2	96.4	100	100	99.2	99.4	26.8	98.2	99.2	94.2	99.6	100

Results: Recognition rate (cheating)

Table 6. Summary of RCI and MLP-F in cheating behaviour simulations with three attributes

Ratio		10%						20%					
Discrimination		Low			High			Low			High		
ATT	Items	RCI	MLPO	MLPT	RCI	MLPO	MLPT	RCI	MLPO	MLPT	RCI	MLPO	MLPT
(1,1,0)	10	0	43.8	91.6	17.2	87.6	100	2.1	46.6	80	39.2	91.2	95.2
	20	7.7	61.8	95.4	22.7	97.8	99.6	22.6	64.4	86.2	100	98.0	99.6
	40	24.7	91.6	99.4	50.6	99.6	100	60.3	95.2	98.8	100	99.8	100
(0,1,1)	10	0.3	29.4	92.2	13.7	84.8	99.8	4.2	27.2	68.6	48.3	88.8	95.2
	20	4.3	53.8	92.6	34.6	97.0	99.6	14.9	56.2	76.8	100	97.4	99.2
	40	25.4	91.6	98.4	65.1	100	100	65.4	94.3	98.2	100	99.6	99.8
(1,0,1)	10	0.2	57.6	94.2	13.5	86.0	98.2	2.5	47.2	84.4	46.1	92.4	97
	20	5.7	60.8	94.8	28.9	98.4	99.8	15.1	58.0	83.4	85.7	99.2	99.6
	40	22.9	94.4	96.8	63.6	100	100	63.8	94.2	98.2	99.8	99.0	100
(1,0,0)	10	1.3	40.6	98.4	12.4	85.2	99.6	5.8	24.4	97	45.7	82.6	100
	20	6.4	51.0	98.8	35.0	80.6	84.8	18.8	33.6	96.2	80.8	85.4	99.8
	40	23.3	66.0	99.6	70.3	95.6	99.6	74.4	57.6	99.6	100	96.2	99.8
(0,1,0)	10	0	40.6	92.8	17.6	85.4	100	5.9	24.6	95.4	24.6	76.2	100
	20	6.1	21.4	99.8	33.6	78.9	99.8	16.3	26.2	96.8	80.4	79.4	99.6
	40	24.1	64.4	99.8	68.2	95.4	99.8	70.3	69.0	99.8	100	97.8	99.8
(0,0,1)	10	0.7	37.4	96.2	4.7	73.6	100	6.7	29.6	95.0	21.9	85.4	99.0
	20	5.9	47.2	98.6	38.9	83.8	99.8	17.2	22.8	94.8	78.0	77.6	99.8
	40	23.9	34.0	99.8	77.2	97.4	100	73.0	82.6	99.8	100	98.4	100
(0,0,0)	10	0	38.2	99.6	7.0	74.6	100	6.3	46.4	98.6	33.5	67.8	100
	20	8.4	72.4	99.8	28.0	92.4	100	18.7	79.2	98.8	77.3	87.6	99.8
	40	25.7	90.4	99.8	65.2	92.4	99.8	68.2	91.0	100	100	97.8	100

Results: Recognition rate (cheating)

Table 7. Summary of RCI and MLP-F in cheating behaviour simulations with six attributes

Ratio		10%						20%					
Discrimination		Low			High			Low			High		
ATT	Items	RCI	MLPO	MLPT	RCI	MLPO	MLPT	RCI	MLPO	MLPT	RCI	MLPO	MLPT
(0,0,0,0,0,0)	20	1.3%	46.6%	99.8%	10.2%	81.8%	100%	15.3%	13.4%	100%	98.9%	75.6%	100%
	40	10.4%	93.0%	99.0%	61.3%	87.8%	100%	58.7%	97.2%	100%	100%	86.8%	100%
(1,0,0,0,0,0)	20	1.7%	29.6%	100%	11.8%	67.4%	100%	7.8%	8.4%	100%	98.1%	65.6%	100%
	40	9.9%	2.6%	99.2%	52.9%	75.8%	100%	55.8%	3.2%	99.8%	100%	70.4%	100%
(0,0,0,0,0,1)	20	0.1%	17.8%	100%	14.2%	67.8%	97.2%	12.2%	17.6%	94.6%	97.0%	57.6%	100%
	40	10.7%	3.5%	99.8%	53.2%	76.2%	100%	65.1%	4.4%	100%	100%	81.6%	100%
(1,1,0,0,0,0)	20	4.4%	23.2%	100%	15.7%	64.8%	100%	13.9%	52.6%	100%	91.1%	64.4%	100%
	40	4.7%	4.4%	100%	38.6%	76.2%	100%	64.0%	2.4%	100%	100%	88.4%	100%
(0,0,0,0,1,1)	20	0.0%	5.2%	99.4%	19.4%	80.8%	100%	11.5%	14.6%	100%	97.5%	87.4%	100%
	40	6.2%	30.6%	100%	43.6%	91.8%	99.2%	37.6%	47.4%	100%	90.5%	94.6%	100%
(1,1,1,0,0,0)	20	0%	27.2%	99.0%	25.8%	72.8%	100%	16.4%	27.8%	100%	84.2%	83.8%	100%
	40	9.6%	8.6%	100%	33.4%	76.4%	100%	57.1%	7.8%	100%	100%	81.2%	100%
(0,0,0,1,1,1)	20	2.4%	31.6%	99.2%	15.6%	82.4%	100%	6.6%	23.6%	100%	73.6%	85.8%	100%
	40	8.5%	49.2%	98.8%	66.5%	96.0%	100%	29.4%	70.8%	100%	93.2%	96.2%	100%
(1,1,1,1,0,0)	20	1.3%	74.4%	100%	11.5%	94.6%	100%	15.7%	70.6%	100%	44.7%	92.2%	100%
	40	9.1%	61.4%	100%	39.5%	94.2%	100%	58.2%	65.2%	100%	100%	95.6%	100%
(0,0,1,1,1,1)	20	2.5%	37.6%	100%	20.4%	94.4%	100%	4.5%	82.8%	100%	70.5%	96.6%	100%
	40	21.3%	92.2%	100%	54.6%	97.4%	100%	30.1%	96.4%	100%	100%	99.6%	100%
(0,1,1,1,1,1)	20	5.1%	53.6%	100%	14.9%	97.2%	100%	1.1%	60.6%	100%	38.8%	98.8%	100%
	40	13.8%	79.4%	95.4%	59.3%	79.0%	100%	17.7%	86.4%	97.0%	93.0%	70.4%	100%

Results: Recognition rate (guessing)

Table 8. Summary of RCI and MLP-F in random guessing simulations

Discrimination		Low			High		
Attributes	Items	RCI	MLPO	MLPT	RCI	MLPO	MLPT
3	10	0.7	64.8	84.4	13.5	56.8	99.6
	20	2.9	67.2	83.4	58.1	86.6	99.6
	40	4.7	89.2	92.4	74.5	97.8	99.4
6	20	1.6	65.2	90.4	28.4	85.4	98.2
	40	6.3	93.2	100	53.8	97.4	99.8

Discussion

- MLP-F has great potential to improve power in detecting aberrant responses for **short exams**
- Generalize the new method to **CD-CAT** using ML is possible
- Consider **other kinds and more complex** aberrant response behaviours
- Generalize this method to many **other CDMs**

The End. Thanks for Listening!

beijing normal university

谢谢大家

多谢晒~

ありがとう

Danke

Merci

Reporter: Yingshi Huang