Logistic Regression

Luca Martino

Introduction

DATA:

$x_k \in \mathbb{R}$ it could be easily generalized for $\mathbf{x}_k \in \mathbb{R}^D$

$y_k \in \{0, 1\}$

$\{x_k, y_k\}_{k=1}^N$

We model directly *the probability of an input x belonging to a class.*

Introduction

We model directly the probability of an input x belonging to a class. In this binary classification case, we focus on the probability of belonging to the class labelled with the label "y_k=1".

Model [edit]

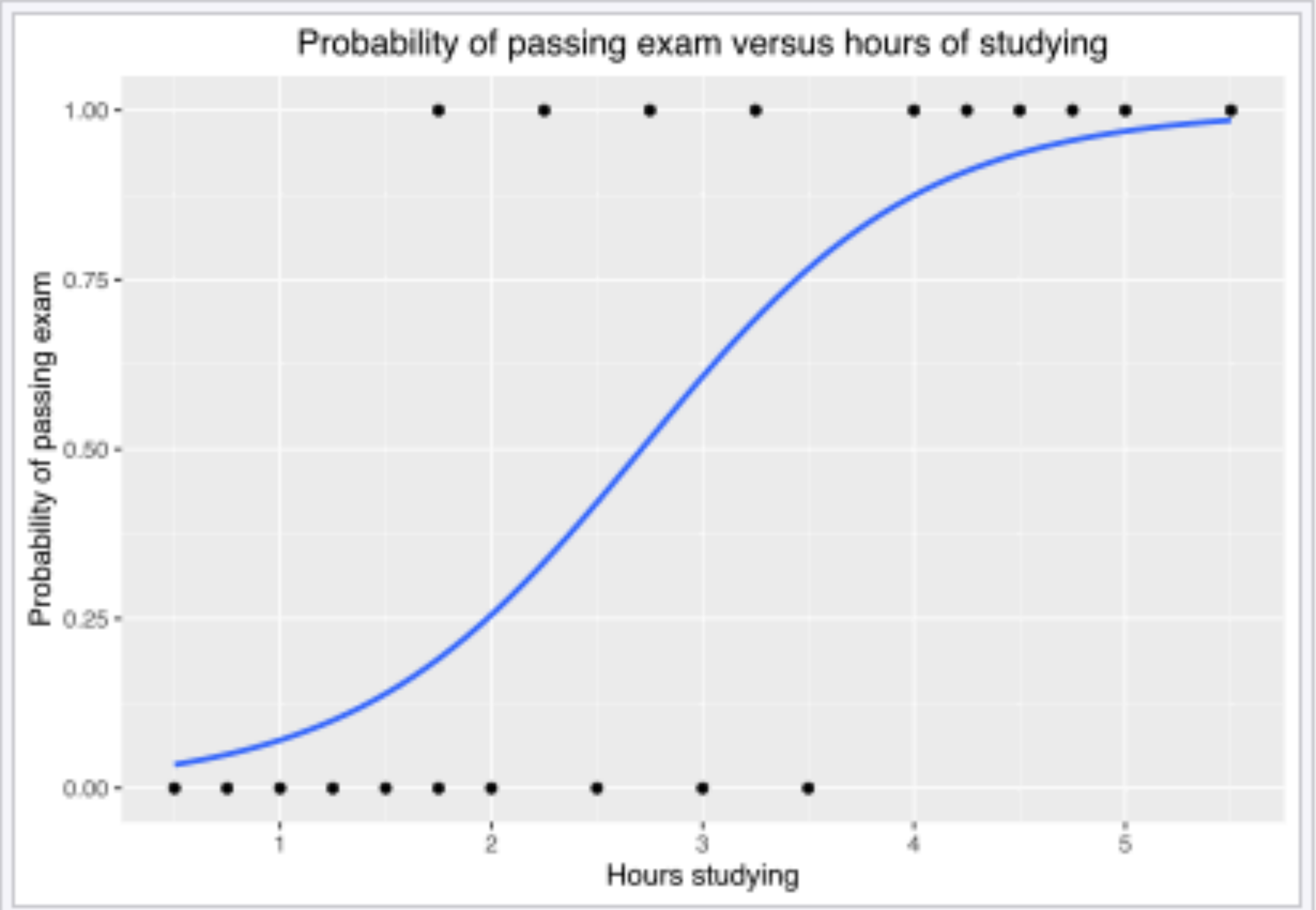
The logistic function is of the form:

$$p(y=1|x) = \frac{1}{1+e^{-(x-\mu)/s}}$$

where μ is a location parameter (the midpoint of the curve, where $p(\mu)=1/2$) and s is a scale parameter. This expression may be rewritten as:

$$p(y=1|x) = \frac{1}{1+e^{-(\beta_0+\beta_1 x)}}$$

where $\beta_0 = -\mu/s$ and is known as the intercept (it is the *vertical* intercept or *y*-intercept of the line $y = \beta_0 + \beta_1 x$), and $\beta_1 = 1/s$ (inverse scale parameter or rate parameter): these are the *y*-intercept and slope of the log-odds as a function of *x*. Conversely, $\mu = -\beta_0/\beta_1$ and $s = 1/\beta_1$.



Equivalent expressions

$$p(y_k = 1|x_k) = p_k = \frac{1}{1 + e^{-(x_k - \mu)/s}}$$

where $\beta_0 = -\mu/s$ and is known as the intercept (it is the *vertical* intercept or *y*-intercept of the line $y = \beta_0 + \beta_1 x$), and $\beta_1 = 1/s$ (inverse scale parameter or rate parameter): these are the *y*-intercept and slope of the log-odds as a function of *x*. Conversely, $\mu = -\beta_0/\beta_1$ and $s = 1/\beta_1$.

$$p(y_k = 1|x_k) = p_k = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_k)}}$$

$$p(y_k = 1|x_k) = p_k = \frac{e^{\beta_0 + \beta_1 x_k}}{1 + e^{\beta_0 + \beta_1 x_k}}$$

Binary case: likelihood function

$$p(y_k = 1|x_k) = p_k = \frac{1}{1 + e^{-(x_k - \mu)/s}}$$

$$p(\mathbf{y}|\mathbf{x}) = p(y_1, ..., y_N|x_1, ..., x_N) = \prod_{k:y_k=1} p_k \prod_{k:y_k=0} (1 - p_k)$$

If y_k=0,1, we can rewrite:

$$p(\mathbf{y}|\mathbf{x}) = p(y_1, ..., y_N|x_1, ..., x_N) = \prod_{k=1}^{N} p_k^{y_k} (1 - p_k)^{1-y_k}$$

Binary case: likelihood function

If y_k=0,1, we can rewrite:

$$p(\mathbf{y}|\mathbf{x}) = p(y_1, ..., y_N | x_1, ..., x_N) = \prod_{k=1}^{N} p_k^{y_k} \left(1 - p_k\right)^{1-y_k}$$

$$\log p(\mathbf{y}|\mathbf{x}) = \sum_{k=1}^{N} \left[\log\left[p_k^{y_k}\right] + \log[\left(1 - p_k\right)^{1-y_k}]\right]$$

In the code that I sent in Studium, I used this formula in order to avoid numerical issues, such as NaN for "0 times -Inf" ="0 x -Inf".

$$\log p(\mathbf{y}|\mathbf{x}) = \sum_{k=1}^{N} [y_k \log\left(p_k\right) + \left(1 - y_k\right) \log\left(1 - p_k\right)]$$

First generalization: With multimensional inputs (straightforward)

Luca Martino

With multimensional inputs

DATA:

$$\mathbf{x}_k = [x_{k,1}, \dots, x_{k,D}]^\top \in \mathbb{R}^D$$

$$y_k \in \{0, 1\}$$

$$\{\mathbf{x}_k, y_k\}_{k=1}^N$$

With multimensional inputs

Then we consider:

$$p(y_k = 1|\mathbf{x}_k) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_{k,1} + \beta_2 x_{k,2} + \ldots + \beta_D x_{k,D})}}$$

$$p(y_k = 1|\mathbf{x}_k) = \frac{1}{1 + e^{-\left(\beta_0 + \sum_{d=1}^{D} \beta_d x_{k,d}\right)}}$$

Here we have to learn all the betas !!!!

Simplifying the previous expressions “using vectors”

RE-DEFINING:

$$\mathbf{x}_k = [{\color{red}1}, x_{k,1}, ..., x_{k,D}]^\top \in \mathbb{R}^{\color{red}D+1}$$

AND:

$$\boldsymbol{\beta} = [\beta_0, \beta_1, ..., \beta_D]$$

then:

$$p_k = p(y_k = 1|\mathbf{x}_k) = \frac{1}{1 + e^{-(\boldsymbol{\beta}\mathbf{x}_k)}} = \frac{e^{\boldsymbol{\beta}\mathbf{x}_k}}{1 + e^{\boldsymbol{\beta}\mathbf{x}_k}}$$

With multimensional inputs

The rest remains the same….same likelihood function:

$$p_k = p(y_k = 1|\mathbf{x}_k)$$

$$p(\mathbf{y}|\mathbf{X}) = p\left(y_1, \ldots, y_N|\mathbf{x}_1, \ldots, \mathbf{x}_N\right) = \prod_{k=1}^{N} p_k^{y_k}\left(1 - p_k\right)^{1-y_k}$$

etc….

Second generalization:
More than 2 classes….

"**Multinomial logistic regression:** Many explanatory variables (inputs) and many categories (outputs, more than 2 classes)"

Luca Martino

Dealing with *M* classes

DATA:

$$\mathbf{x}_k = \left[x_{k,1}, ..., x_{k,D}\right]^\top \in \mathbb{R}^D$$

$$y_k \in \{0, 1, 2, ..., M-1\}$$

$$\{\mathbf{x}_k, y_k\}_{k=1}^N$$

Dealing with *M* classes

Again, we model directly *the probability of an input x belonging to a class.*

RE-DEFINING:

$$\mathbf{x}_k = [1, x_{k,1}, ..., x_{k,D}]^\top \in \mathbb{R}^{D+1}$$

AND:

$$\boldsymbol{\beta}_m = [\beta_{m,0}, \beta_{m,1}, ..., \beta_{m,D}], \qquad \text{with } m = 1, ..., M-1$$

then:

$$p_{k,m} = p(y_k = m|\mathbf{x}_k) = \frac{e^{\boldsymbol{\beta}_m \mathbf{x}_k}}{1 + \sum_{j=1}^{M-1} e^{\boldsymbol{\beta}_j \mathbf{x}_k}}$$

$$p_{k,0} = p(y_k = 0|\mathbf{x}_k) = 1 - \sum_{m=1}^{M-1} \frac{e^{\boldsymbol{\beta}_m \mathbf{x}_k}}{1 + \sum_{j=1}^{M-1} e^{\boldsymbol{\beta}_j \mathbf{x}_k}}$$

Dealing with *M* classes

then:

$$p_{k,m} = p(y_k = m|\mathbf{x}_k) = \frac{e^{\boldsymbol{\beta}_m \mathbf{x}_k}}{1 + \sum_{j=1}^{M-1} e^{\boldsymbol{\beta}_j \mathbf{x}_k}}$$

$$p_{k,0} = p(y_k = 0|\mathbf{x}_k) = 1 - \sum_{m=1}^{M-1} \frac{e^{\boldsymbol{\beta}_m \mathbf{x}_k}}{1 + \sum_{j=1}^{M-1} e^{\boldsymbol{\beta}_j \mathbf{x}_k}} = 1 - \frac{\sum_{m=1}^{M-1} e^{\boldsymbol{\beta}_m \mathbf{x}_k}}{1 + \sum_{j=1}^{M-1} e^{\boldsymbol{\beta}_j \mathbf{x}_k}}$$

$$= \frac{1 + \cancel{\sum_{j=1}^{M-1} e^{\boldsymbol{\beta}_j \mathbf{x}_k}} - \cancel{\sum_{m=1}^{M-1} e^{\boldsymbol{\beta}_m \mathbf{x}_k}}}{1 + \sum_{j=1}^{M-1} e^{\boldsymbol{\beta}_j \mathbf{x}_k}}$$

$$= \frac{1}{1 + \sum_{j=1}^{M-1} e^{\boldsymbol{\beta}_j \mathbf{x}_k}}$$

Dealing with *M* classes

then, FINALLY:

$$p_{k,m} = p(y_k = m|\mathbf{x}_k) = \frac{e^{\boldsymbol{\beta}_m \mathbf{x}_k}}{1 + \sum_{j=1}^{M-1} e^{\boldsymbol{\beta}_j \mathbf{x}_k}}$$

$$p_{k,0} = p(y_k = 0|\mathbf{x}_k) = \frac{1}{1 + \sum_{j=1}^{M-1} e^{\boldsymbol{\beta}_j \mathbf{x}_k}}$$

Dealing with *M* classes

OR MORE GENERALLY FOR a generic x (test input):

$$p(y_k = m|\mathbf{x}) = \frac{e^{\boldsymbol{\beta}_m \mathbf{x}}}{1 + \sum_{j=1}^{M-1} e^{\boldsymbol{\beta}_j \mathbf{x}}}$$

$$p(y_k = 0|\mathbf{x}) = 1 - \sum_{m=1}^{M-1} p(y_k = m|\mathbf{x}) = \frac{1}{1 + \sum_{j=1}^{M-1} e^{\boldsymbol{\beta}_j \mathbf{x}}}$$

Dealing with *M* classes

WE HAVE TO LEARN (M-1) VECTORS OF BETAs….

hence, D *x* (M-1) scalar numbers to learn !!!!

$$\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_{M-1}$$

Example of construction of the likelihood function

$$\text{Classes} = 0, 1, 2, 3 \implies M = 4 \qquad N = 5$$

$\mathbf{x}_1$	$\mathbf{x}_2$	$\mathbf{x}_3$	$\mathbf{x}_4$	$\mathbf{x}_5$
$\mathbf{y}_1 = 3$	$\mathbf{y}_2 = 3$	$\mathbf{y}_3 = 1$	$\mathbf{y}_4 = 0$	$\mathbf{y}_5 = 2$

$$p(\mathbf{y}|\mathbf{X}) = p_{1,3} \times p_{2,3} \times p_{3,1} \times p_{4,0} \times p_{5,2}$$

$$p_{4,0} = 1 - p_{4,1} - p_{4,2} - p_{4,3}$$

Isotopic multi-output logistic regression for parallel classification problems

Luca Martino

DATA (multioutput - isotopic scenario)

OUTPUTS - isotopic scenario

$$\begin{array}{c|ccccccc} & & & & & & & j \\ \hline x_1 & y_{1,1} & y_{1,2} & y_{1,3} & \cdots & \cdots & \cdots & y_{1,N} \\ x_2 & y_{2,1} & y_{2,2} & y_{2,3} & \cdots & \cdots & \cdots & y_{2,N} \\ & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_M & y_{M,1} & y_{M,2} & y_{M,3} & \cdots & \cdots & \cdots & y_{M,N} \\ i & & & & & & & \end{array}$$

$$\boxed{y_{i,j} \in \{0, 1\}}$$

isotopic scenario:

N outputs share the same input x

DATA points (*M*):

$$\{\mathbf{x}_m, \mathbf{y}_m\}_{m=1}^{M}$$

$$\mathbf{y}_m = [y_{m,1}, y_{m,2}, \ldots, y_{m,N}]$$

$$i = 1, 2, \ldots, M$$

$$j = 1, 2, \ldots, N$$

About the notation

A MORE PROPER NOTATION FOR THESE SLIDES SHOULD BE OBTAINED SWITCHING *M* AND *N*:

- ***N* should be the number of data points**
- ***and M* should be the number of outputs per each input *x***

However, we have used this notation for linking this part with the slides on ITEM RESPONSE THEORY (where we use exactly the notation employed here).

DATA points (*M*):

$$\{\mathbf{x}_m, \mathbf{y}_m\}_{m=1}^{M}$$

$$\mathbf{y}_m = [y_{m,1}, y_{m,2}, ..., y_{m,N}]$$

$$\boxed{y_{i,j} \in \{0, 1\}}$$

$$i = 1, 2, ..., M$$

$$j = 1, 2, ..., N$$

About the notation

DATA points (*M*):

$$\{\mathbf{x}_m, \mathbf{y}_m\}_{m=1}^{M}$$

$$\mathbf{X} = [\mathbf{x}_1, ..., \mathbf{x}_M]$$

$$\mathbf{y}_m = [y_{m,1}, y_{m,2}, ..., y_{m,N}]$$

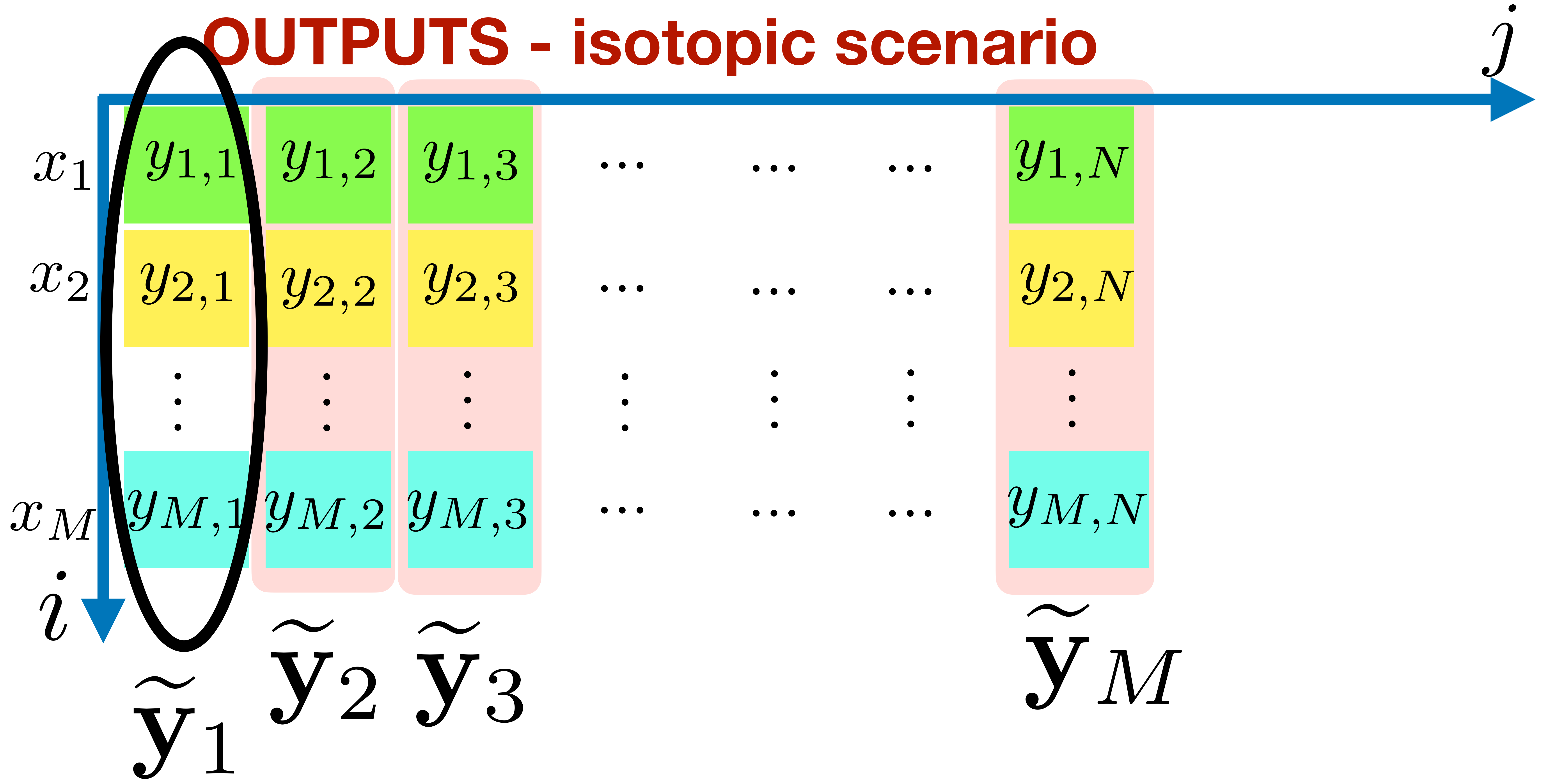
Other vision of the data ("vertical vision"):

$$\{x_m, y_{m,j}\}_{m=1}^{M}$$

$$\widetilde{\mathbf{y}}_j = [y_{1,j}, y_{2,j}, ..., y_{M,j}]^{\top}$$

Other vision of the data

Other vision of the data ("vertical vision"):



But actually what can we do with this data?

For simplicity, considering scalar inputs x…just for simplicity and facilitate the comparison with the IRT:

$$x_m \in \mathbb{R} \qquad \{x_m, y_{m,j}\}_{m=1}^{M} \qquad \mathbf{X} = [x_1, ..., x_M]$$

We have N-parallel classification problems each one with likelihood:

$$p\left(y_{m,j} = 1 | x_m\right) = p_{m,j} = \frac{1}{1 + e^{-(x_m - \mu_j)/s_j}}$$

$$p(\widetilde{\mathbf{y}}_j | \mathbf{X}) = p\left(y_{1,j}, \ldots, y_{M,j} | x_1, \ldots, x_M\right) = \prod_{m:y_{m,j}=1} p_{m,j} \prod_{m:y_{m,j}=0} \left(1 - p_{m,j}\right)$$

$$p\left(y_{1,j}, \ldots, y_{M,j} | x_1, \ldots, x_M\right) = \prod_{m=1}^{M} p_{m,j}^{y_{m,j}} \left(1 - p_{m,j}\right)^{(1 - y_{m,j})}$$

But actually what can we do with this data?

We have N-parallel classification problems each one with likelihood function. They share the same *M* inputs.

We have to find, in this case with scalar inputs, *N* different pairs of mu and s, one for each parallel classification problems.

$$p\left(y_{m,j} = 1 | x_m\right) = p_{m,j} = \frac{1}{1 + e^{-(x_m - \mu_j)/s_j}}$$

$$\mu_j \,, s_j \quad \text{for } j = 1, \ldots, N$$