Machine Learning (Lab support)

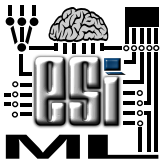
Multi-class and multi-label classification

Abdelkrime Aries

Laboratoire de la Communication dans les Systèmes Informatiques (LCSI)
École nationale Supérieure d'Informatique (ESI, ex. INI), Algiers, Algeria

Academic year: 2024-2025






Attribution 4.0 International (CC BY 4.0)

https://creativecommons.org/licenses/by/4.0/deed.en

You are free to:

Share — copy and redistribute the material in any medium or format

Adapt — remix, transform, and build upon the material for any purpose, even commercially.

The licensor cannot revoke these freedoms as long as you follow the license terms.

Under the following terms:

Attribution — You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.

No additional restrictions — You may not apply legal terms or technological measures that legally restrict others from doing anything the license permits.

Machine Learning (Lab support)

Multi-class/label: Introduction

- We have already seen ...
  - how to estimate the probability of a class
  - using logistic regression
  - in case of binary classification (belongs to a class or not)
- However ...
  - how to estimate them in case of multiple classes?
  - how to assign a sample into many classes at once?

Machine Learning (Lab support)

Multi-class/label: Plan

Section 1

Classification

Multi-class/label

Classification

Classification

Binary

Classes: 1

Choice: 1

Multi-class

One-Output

Classes: L >= 2

Choice: 1

Multi-label

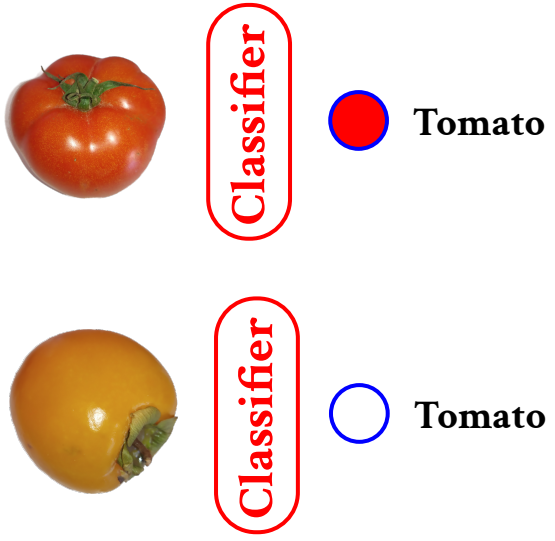
Classes: L >= 2

Choice: 0 <= K <= L

Multi-class/label: Classification

Binary classification

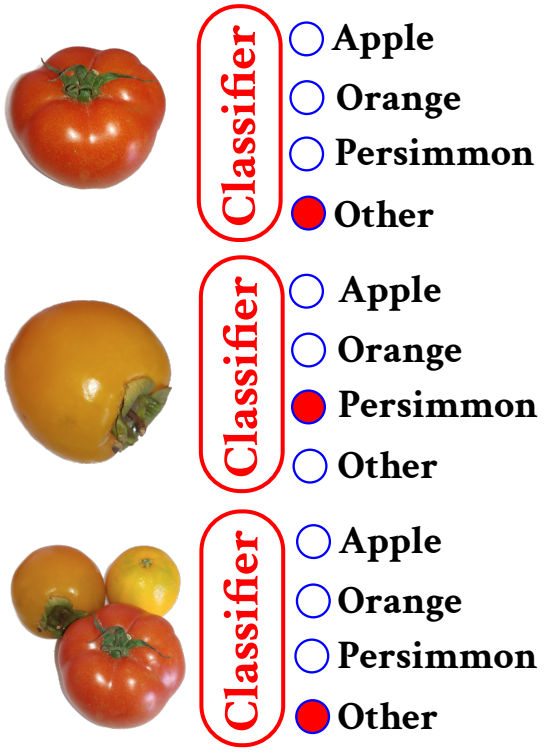
- Number of available classes: 2 (Actually, it is just one class. We test if a sample belongs to it or not)
- Number of chosen classes: 1
- Examples
  - Classify an image as containing a specific object or not.
  - Finding out if a message is a spam or not.



Multi-class/label: Classification

Multi-class classification

- Number of available classes: L $\geq$ 2
- Number of chosen classes: 1
- Examples
  - Detect an animal from an input image.
  - Detect a movie's certification (PG, R, etc.) from its description.



Multi-class/label: Classification

Multi-class classification: Methods

Multi-class with one output

Generalization

Adapt algorithms to accept multiple classes.
Naive bayes and decision trees are multi-class by default.
Logistic regression: using softmax with a generalized loss function.

Using binary models

One-vs-Rest

Train L binary models.
Each is trained on its samples vs the rest of samples.
To estimate the class, use the most probable one (Argmax)

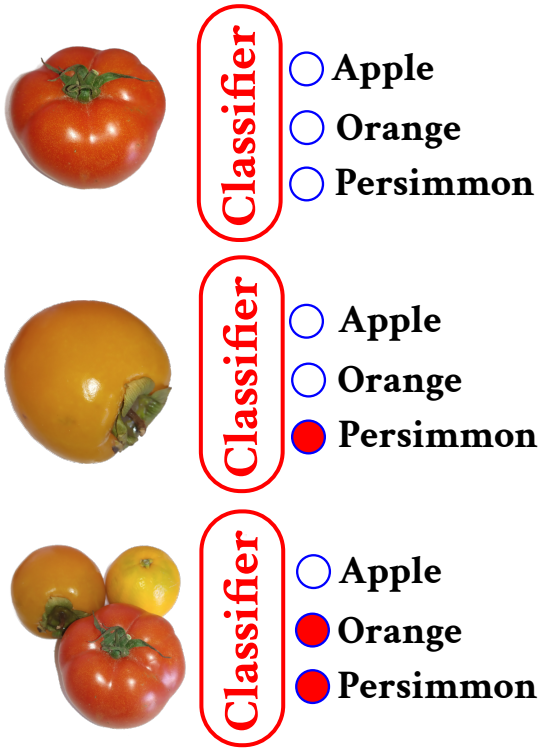
One-vs-One

Train L*(L-1)/2 binary models.
Each is trained on its samples vs another class's samples.
To estimate the class, use the majority vote

Multi-class/label: Classification

Multi-label classification

- Number of available classes: $L \geq 2$
- Number of chosen classes: $K \leq L$
- Examples
  - Detect many animals from an input image.
  - Find the clothes (hat, jeans, scarf, etc.) from an image.
  - Detect movie's genres (Sci-fi, Action, Comedy, etc.) from its description.



Multi-class/label: Classification

Multi-label classification: Methods [Madjarov et al., 2012]

Multi-label

Algorithm adaptation

Adapt algorithms to accept multiple labels. For example decision trees can be modified to accept many classes in the leafs.

Algorithm transformation

Binary relevance

OvR classifiers each estimates a class apart

Pair-wise

OvR classifiers using cumulated votes to decide relevant classes

Label powerset

Multi-class with all combinations as classes

Ensemble based

Like ensemble methods but using a threshold over cumulative votes to select a class

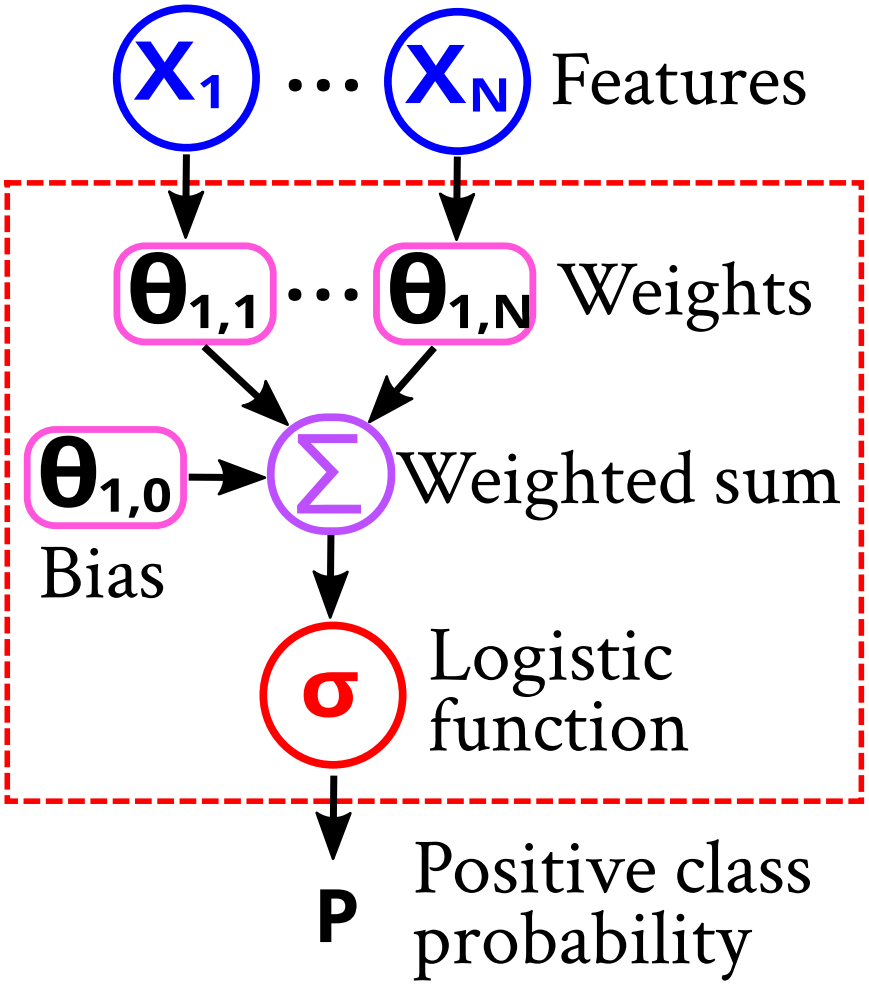
- scikit-multilearn: http://scikit.ml/

Section 2

Binary logistic regression

Multi-class/label

Binary logistic regression



Multi-class/label: Binary logistic regression

Probability estimation

$$Z = \sum_{j=1}^{N} \theta_j X_j = X \cdot \theta$$

- $Z[M]$ is a vector of M elements (samples)
- $X[M, N]$ is a matrix of M samples and N features
- $\theta[N]$ is a vector of N parameters (weights)

$$H = \sigma(Z) = \frac{1}{1 + e^{-Z}}$$

- $H[M]$ is a vector of M probabilities (samples)

Multi-class/label: Binary logistic regression

Cost and Gradient

$$J_\theta = BCE = \frac{-1}{M}\sum_{i=1}^{M}[Y^{(i)}\log(H^{(i)}) + (1-Y^{(i)})\log(1-H^{(i)})]$$

- $Y[M]$ and $H[M]$ are two vectors of M elements (samples)
- J_θ is a scalar

$$\frac{\partial BCE}{\partial \theta_j} = \frac{1}{M}\sum_{i=1}^{M}(H^{(i)} - Y^{(i)})X_j^{(i)}$$

$$\frac{\partial BCE}{\partial \theta} = \frac{1}{M}(H-Y)\cdot X$$

- $\frac{\partial BCE}{\partial \theta}[N]$ is a vector of N elements (features)

Multi-class/label: Binary logistic regression

Gradient (derivation)

$$\frac{\partial BCE}{\partial \theta_j} = \frac{-1}{M}\sum_{i=1}^{M}\frac{\partial}{\partial \theta_j}[Y^{(i)}\log(H^{(i)}) + (1-Y^{(i)})\log(1-H^{(i)})]$$

$$= \frac{-1}{M}\sum_{i=1}^{M}[Y^{(i)}\frac{\partial}{\partial \theta_j}\log(H^{(i)}) + (1-Y^{(i)})\frac{\partial}{\partial \theta_j}\log(1-H^{(i)})]$$

$$= \frac{-1}{M}\sum_{i=1}^{M}[Y^{(i)}\frac{1}{H^{(i)}}\frac{\partial}{\partial \theta_j}H^{(i)} + (1-Y^{(i)})\frac{-1}{1-H^{(i)}}\frac{\partial}{\partial \theta_j}H^{(i)})]$$

$$= \frac{-1}{M}\sum_{i=1}^{M}\frac{Y^{(i)}-H^{(i)}}{H^{(i)}(1-H^{(i)})}\frac{\partial}{\partial \theta_j}H^{(i)}$$

$$\frac{\partial H^{(i)}}{\partial \theta_j} = \frac{\partial \sigma(Z^{(i)})}{\partial Z^{(i)}}\frac{\partial Z^{(i)}}{\partial \theta_j} = [\sigma(Z^{(i)})(1-\sigma(Z^{(i)}))]\frac{\partial}{\partial \theta_j}\sum_{k=0}^{N}\theta_k X_k^{(i)} = H^{(i)}(1-H^{(i)})X_j^{(i)}$$

$$\frac{\partial BCE}{\partial \theta_j} = \frac{-1}{M}\sum_{i=1}^{M}\frac{Y^{(i)}-H^{(i)}}{H^{(i)}(1-H^{(i)})}[H^{(i)}(1-H^{(i)})X_j^{(i)}]$$

$$= \frac{1}{M}\sum_{i=1}^{M}(H^{(i)}-Y^{(i)})X_j^{(i)}$$

Multi-class/label: Binary logistic regression

Parameters' update

$$\theta = \theta - \alpha \frac{\partial J_\theta}{\partial \theta}$$

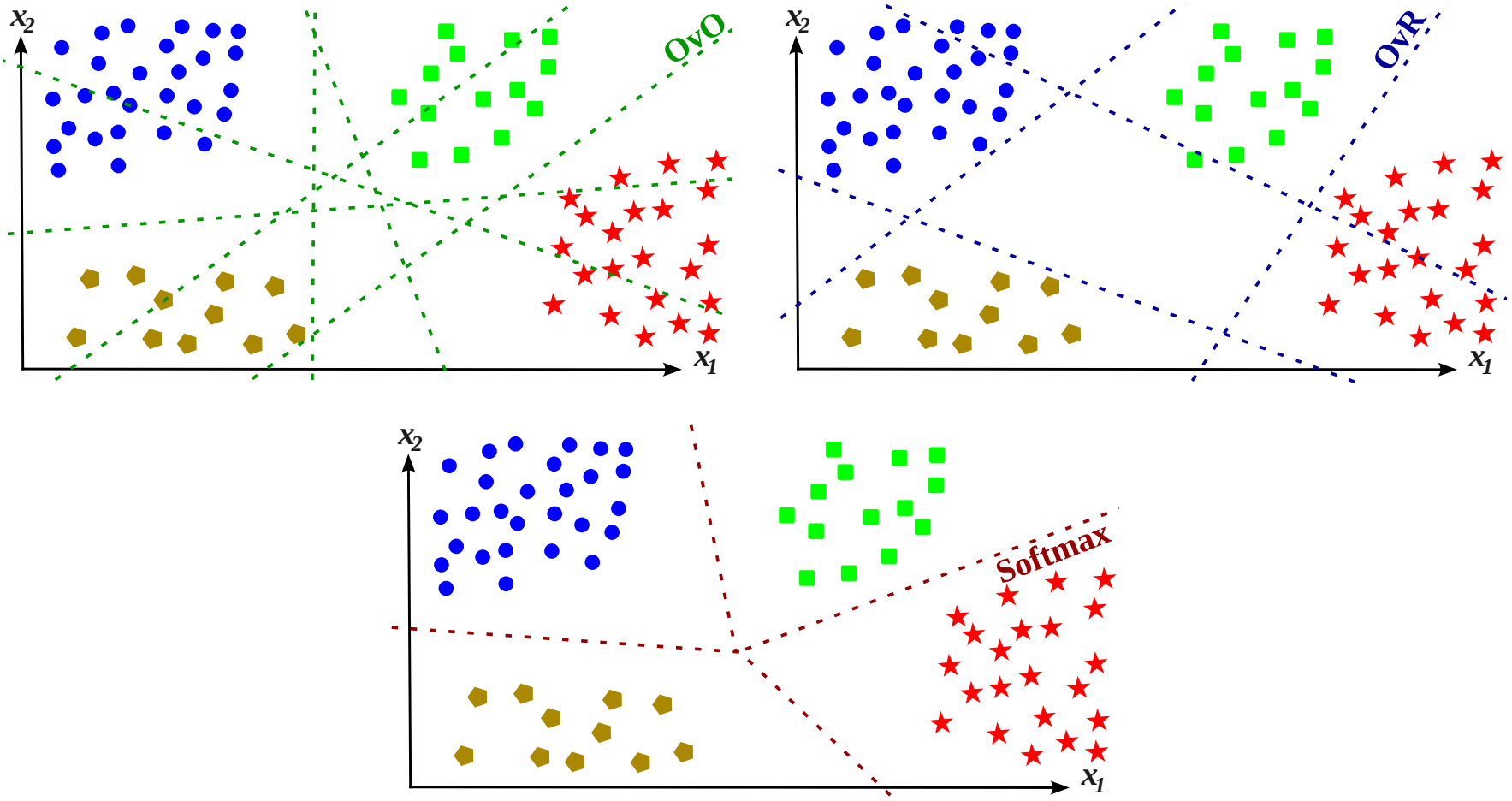
- $\frac{\partial J_\theta}{\partial \theta}[N]$ is a vector of N elements (features)
- $\theta[N]$ is a vector of N elements (features)
- α is a learning rate

Section 3

Multi-class logistic regression

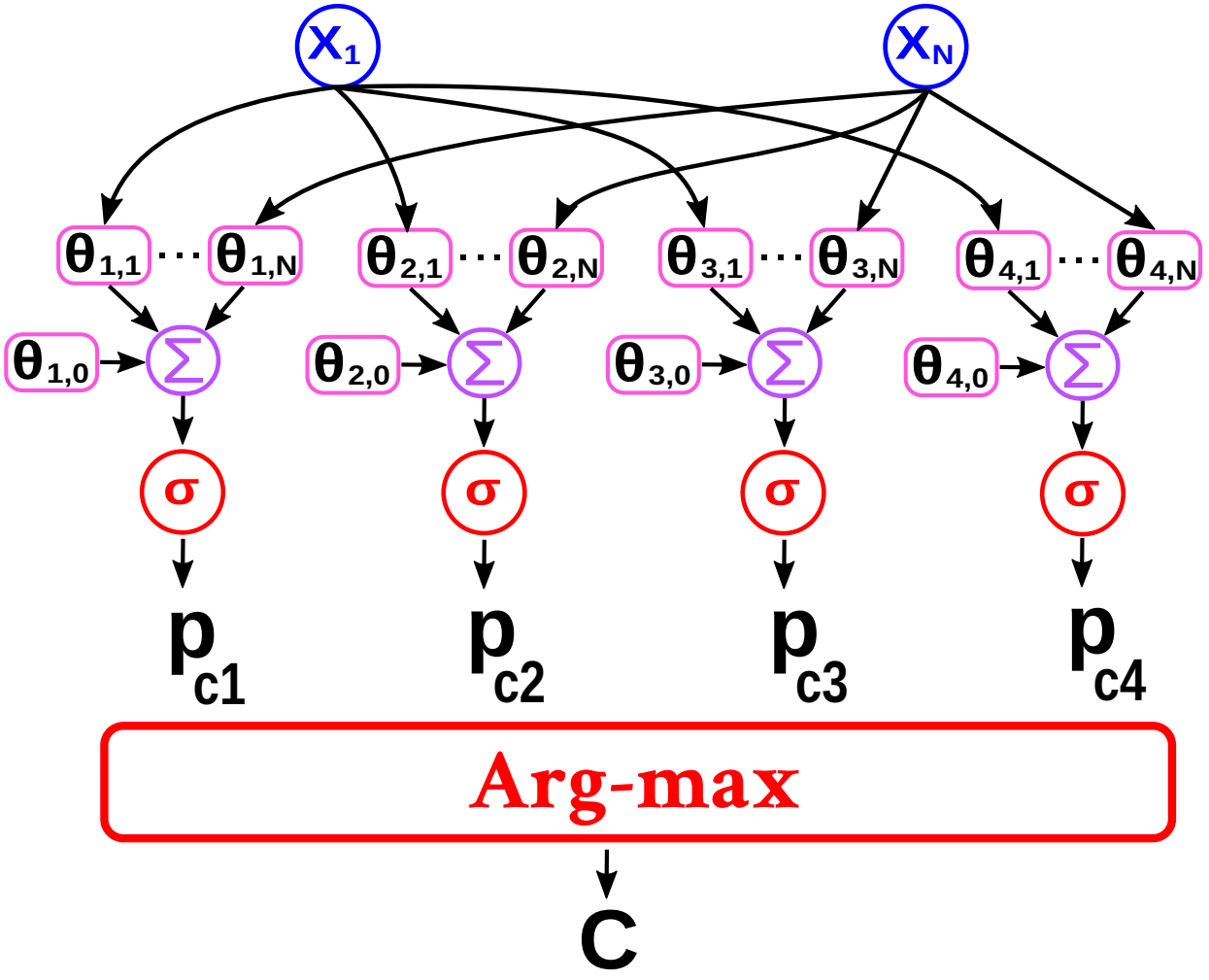
Multi-class/label

Multi-class logistic regression



Multi-class/label: Multi-class logistic regression

One-vs-Rest



Multi-class/label: Multi-class logistic regression

One-vs-Rest: Description

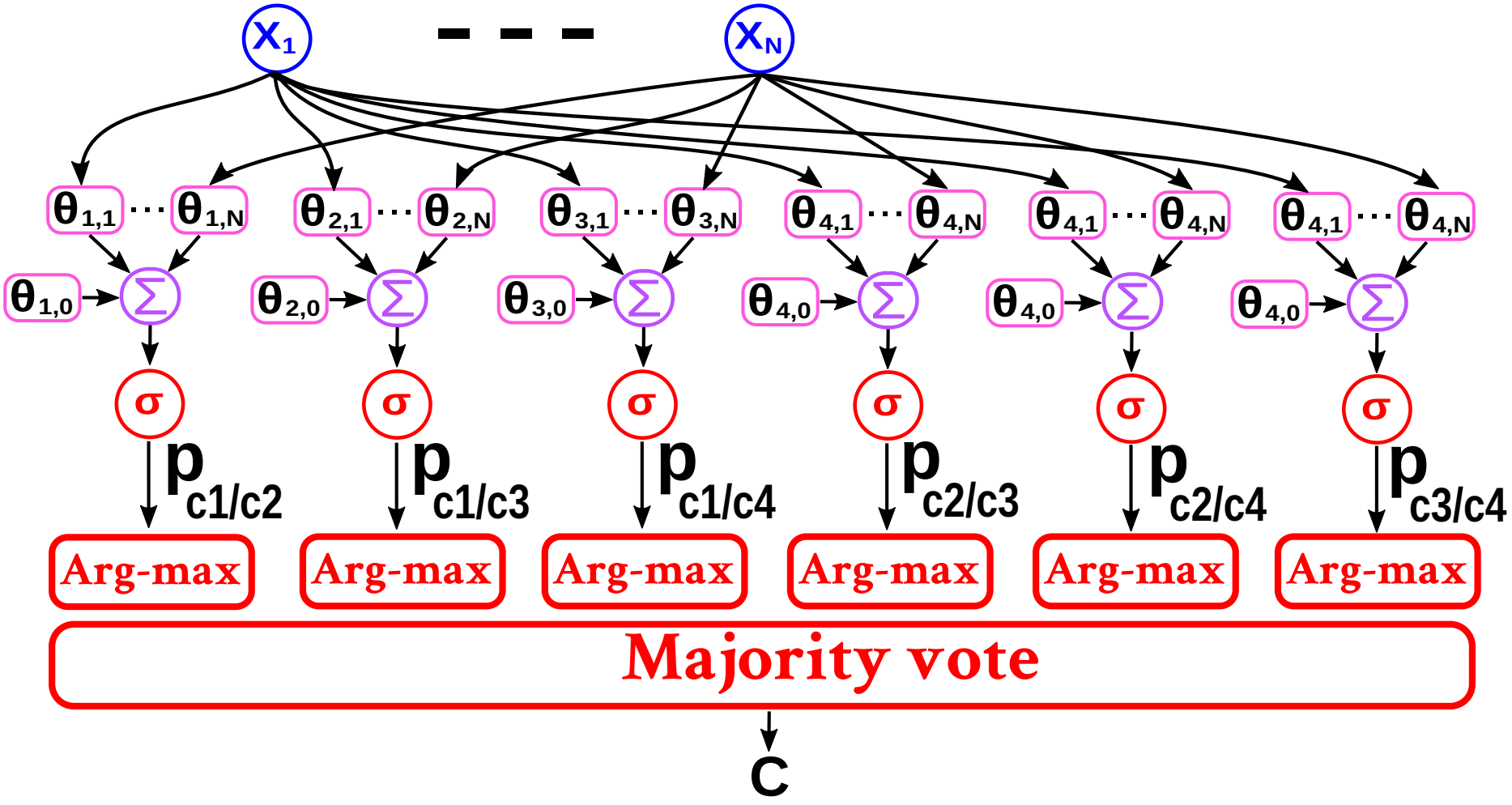
Given L output classes:

- Training
  - For each class C_l, we train a binary model M_l separately
  - In this case, we will have L binary models
  - The positive class is represented by C_l class's samples
  - The negative class is represented by the rest of the samples
- Estimation
  - Given a sample
  - For each class C_l, we estimate its probability using M_l
  - We take the class with the maximum probability
  - In this case, the sum of probabilities does not always give 1

$$\sum_{l=1}^{L} P(l|x;\theta^{M_l}) \in [0, L]$$

Multi-class/label: Multi-class logistic regression

One-vs-One



Multi-class/label: Multi-class logistic regression

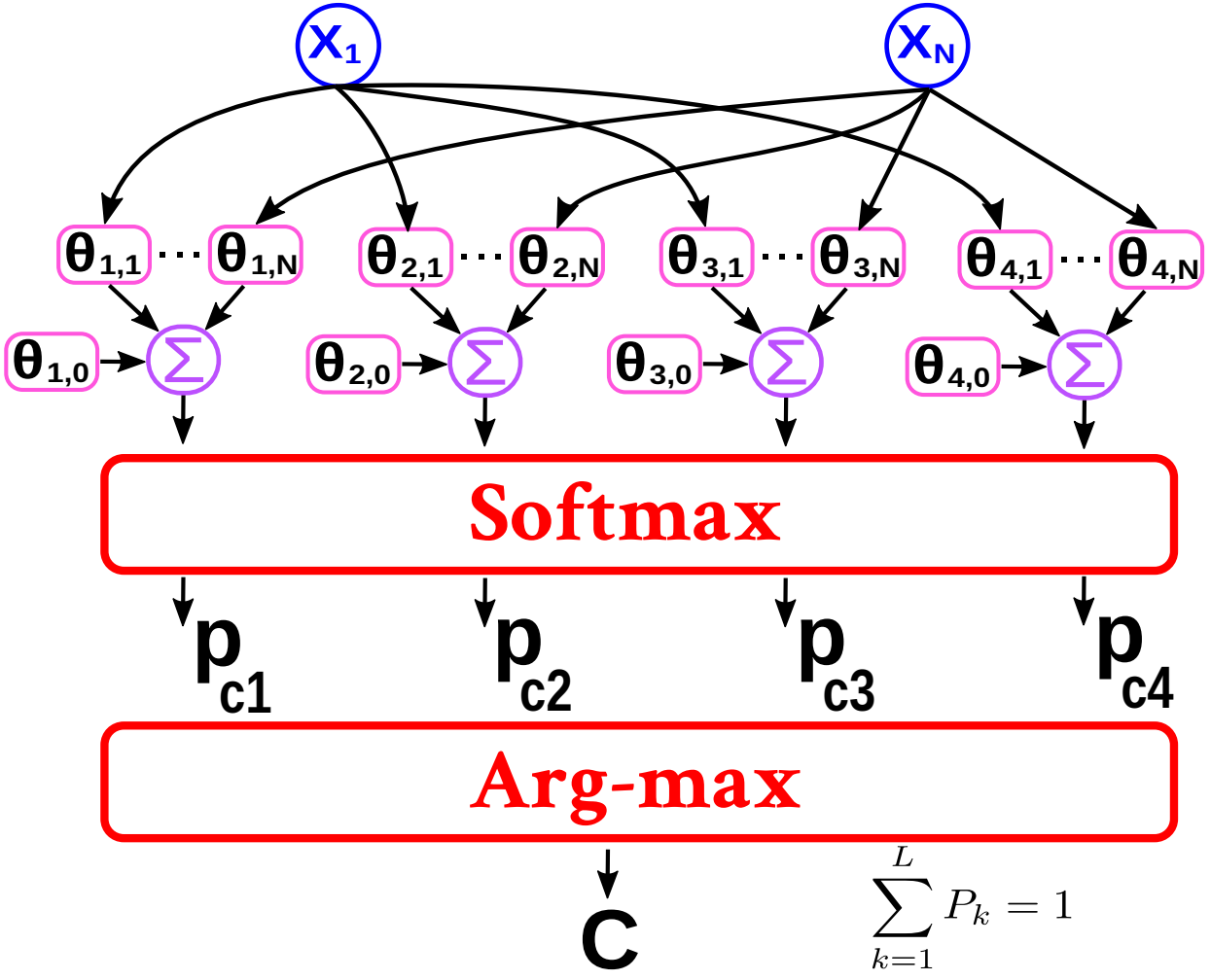
One-vs-One: Description

Given L output classes:

- Training
  - For each two classes C_l and $C_{l'}$, we train a binary model $M_{ll'}$ separately
  - In this case, we will have $L(L-1)/2$ binary models
  - The positive class is represented one of the two classes
  - The negative class is represented by the other class
- Estimation
  - Given a sample
  - For each model $C_{ll'}$, we estimate a probability
  - In this case, the class is either l or l'
  - We count the number of each class being estimated
  - The majority wins (vote)

Multi-class/label: Multi-class logistic regression

Multinomial



$$\sum_{k=1}^{L} P_k = 1$$

Multi-class/label: Multi-class logistic regression

Multinomial: Description

Given L output classes:

- This is a generalization of binary classification in logistic regression
- Training
  - For each class C_l, we calculate a weighted sum
  - We apply **Softmax** function on these outputs
  - It is a function which transforms the sums into probabilities
  - Also, it normalizes these probabilities to get a sum of 1
  - The output classes are represented using One-Hot
  - The cost function is a generalized version of that of binary classification
- Estimation
  - Given a sample
  - We apply the model to get a vector of probabilities
  - The most probable class wins

Multi-class/label: Multi-class logistic regression

Multinomial: Probability estimation

$$Z = \sum_{j=1}^{N} \theta_j X_j = X \cdot \theta$$

- $Z[M, L]$ is a matrix of M samples and L classes
- $X[M, N]$ is a matrix of M samples and N features
- $\theta[N, L]$ is a matrix of N features and L classes

$$H = softmax(Z) = \frac{e^Z}{\sum_{k=1}^{L} e^{Z_k}}$$

- $H[M, L]$ is a matrix of M samples and L classes

Multi-class/label: Multi-class logistic regression

Multinomial: Cost and gradient

$$J_\theta = \frac{-1}{M} \sum_{i=1}^{M} \sum_{k=1}^{L} Y_k^{(i)} \log(H_k^{(i)})$$

- $Y[M, L]$ and $H[M, L]$ are two matrices of $M \times L$ elements (features X classes)
- J_θ is a scalar

$$\frac{\partial J}{\partial \theta_{jk}} = \frac{1}{M} \sum_{i=1}^{M} (H_k^{(i)} - Y_k^{(i)}) X_j^{(i)}$$

$$\frac{\partial J}{\partial \theta} = \frac{1}{M} X^T \cdot (H - Y)$$

- $\frac{\partial J}{\partial \theta}[N, L]$ is a matrix of $N \times L$ elements (features X classes)

Multi-class/label: Multi-class logistic regression

Multinomial: Parameters' update

$$\theta = \theta - \alpha \frac{\partial J_\theta}{\partial \theta}$$

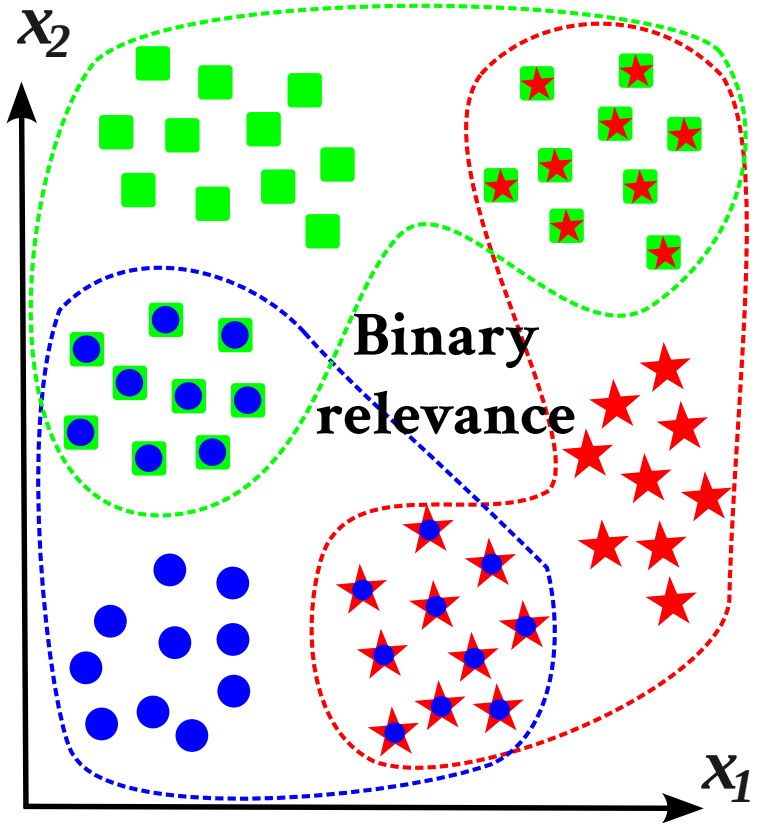
- $\frac{\partial BCE}{\partial \theta}[N, L]$ is a matrix of $N \times L$ elements (features X classes)
- $\theta[N, L]$ is a matrix of N features and L classes
- α is the learning rate

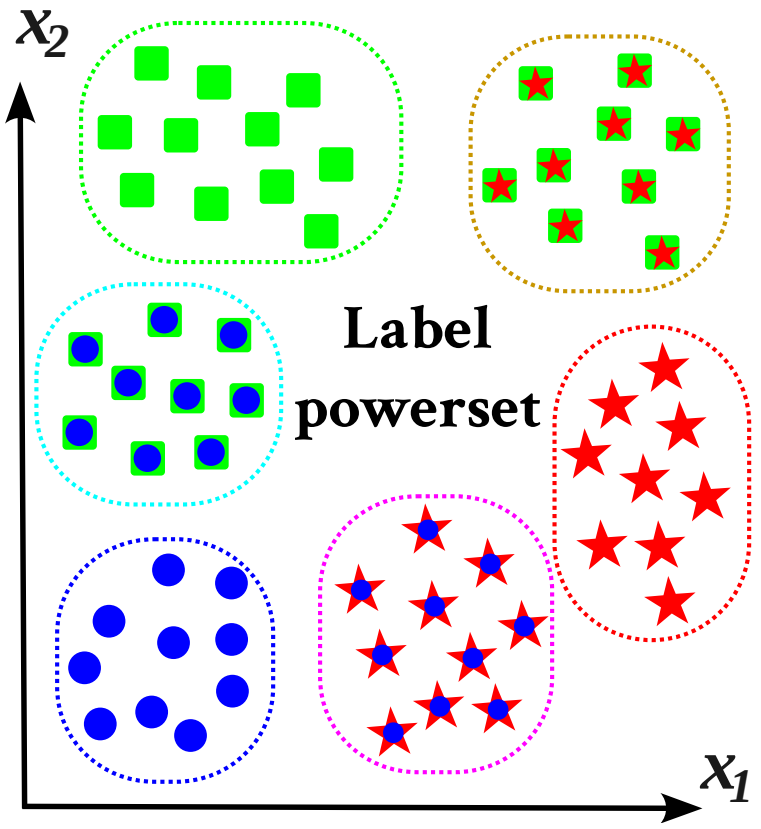
Section 4

Multi-label logistic regression

Multi-class/label

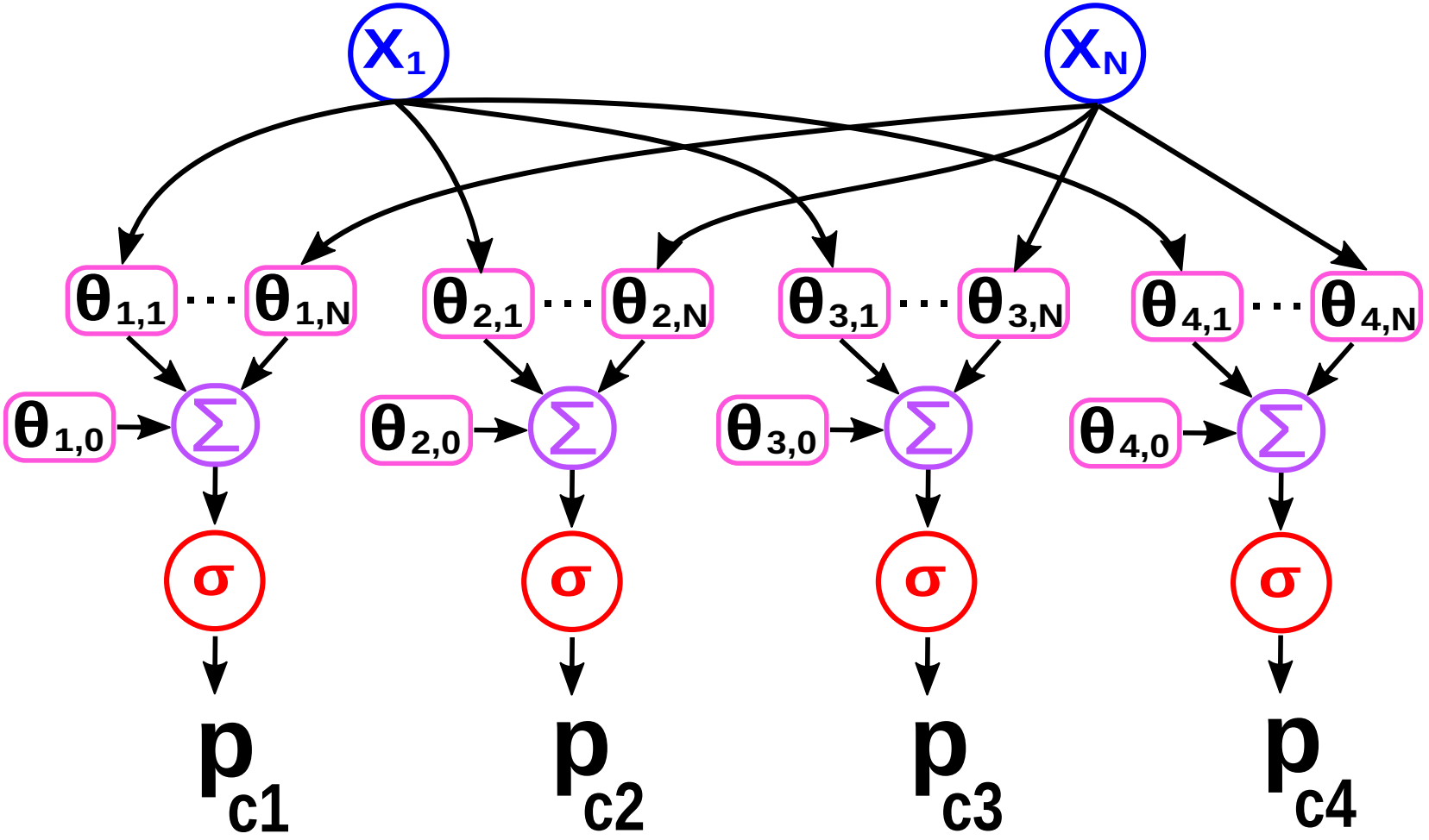
Multi-label logistic regression





Multi-class/label: Multi-label logistic regression

Binary relevance



Multi-class/label: Multi-label logistic regression

Binary relevance: Description

Given L output classes:

- Training
  - For each class C_l, we train a binary model M_l separately
  - In this case, we will have L binary models
  - The positive class is represented by C_l class's samples
  - The negative class is represented by the rest of the samples
- Estimation
  - Given a sample
  - For each class C_l, we estimate its probability using M_l
  - If the probability is greater or equals 50%, then the sample belongs to the class C_l

Multi-class/label: Multi-label logistic regression

Label powerset

Dataset with 3 classes (c1, c2, c3) plus other (o)

- Samples o
- Samples c1
- Samples c1+c2
- Samples c1+c2+c3
- Samples c3
- Samples c2
- Samples c1+c3
- Samples c2+c3

Multi-class classification

(select one of the 8 classes "combinations")

Multi-class/label: Multi-label logistic regression

Label powerset: Description

Given L output classes:

- Training
  - We look for all combinations of classes
  - Each combination is considered as a new class
  - We train a multi-class model
- Estimation
  - Given a sample
  - We use our trained multi-class model to get one class
  - This class is a combination of many original classes

Section 5

Bibliography

Bibliography

Madjarov, G., Kocev, D., Gjorgjevikj, D., and Džeroski, S. (2012).
An extensive experimental comparison of methods for multi-label learning.
Pattern Recognition, 45(9):3084–3104.
Best Papers of Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA'2011).

Keep scrolling

...

Maybe there are some hidden slides :)