CLASSIFICATION
REGRESSION MODELS

Binomial Logistic Regression

Refers binary classification problems to linear regression by exploiting an appropriate data transformation.

Assume that class Y takes values in $\{0,1\}$, then the logistic regression model represents the a posteriori probability of class Y given the vector of explanatory variables $\underline{\boldsymbol{X}}$

$$P(Y \mid \underline{X})$$

through a logistic function

$$P(Y=0 \mid \underline{X}=\underline{x}) = \frac{1}{1+\exp(\underline{w}\cdot\underline{x})} \qquad P(Y=1 \mid \underline{X}=\underline{x}) = \frac{\exp(\underline{w}\cdot\underline{x})}{1+\exp(\underline{w}\cdot\underline{x})}$$

where the vector $\underline{\boldsymbol{w}}$ is the vector of parameters of the logistic regression model and has the same dimension of the vector $\underline{\boldsymbol{X}}$ of the explanatory variables.

Regression Models

The two previous relationships can be appropriately combined to obtain the following relationship

$$\log \frac{P(Y=1|\underline{X}=\underline{x})}{P(Y=0|\underline{X}=\underline{x})} = \underline{w} \cdot \underline{x}$$

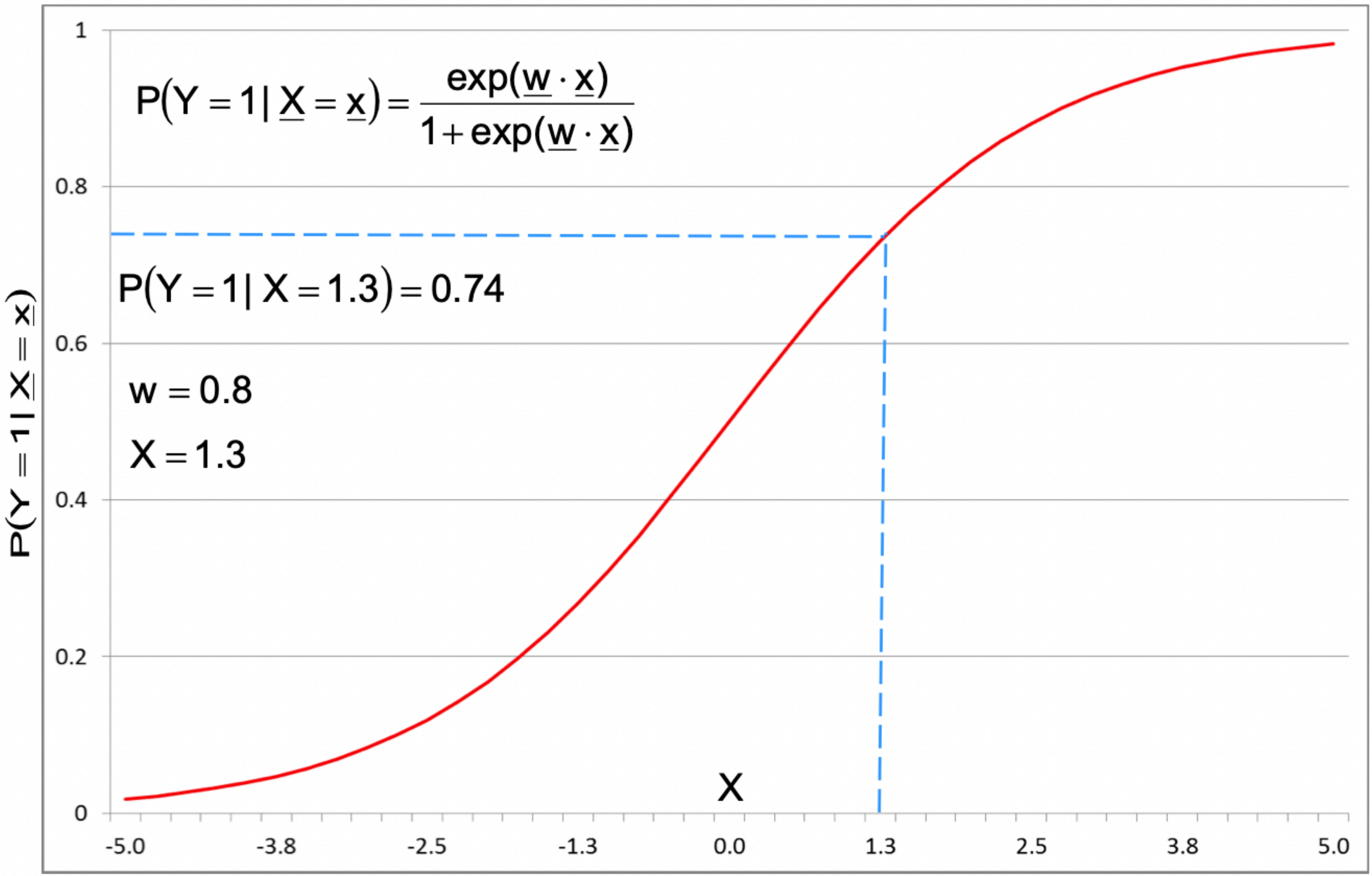
Therefore, if we place

$$Z = \log \frac{P(Y=1|\underline{X}=\underline{x})}{P(Y=0|\underline{X}=\underline{x})}$$

The problem of binary classification is brought back to that of the identification of a linear regression model between the dependent variable Z and the explanatory variables $\underline{\boldsymbol{X}}$.

After determining the coefficients of the regression model and verifying its significance, the anti-transformation of variable Z is applied in order to subsequently use the model for forecasting purposes on any new instance $\underline{\boldsymbol{x}}$ of the vector of explanatory variables.

Regression Models



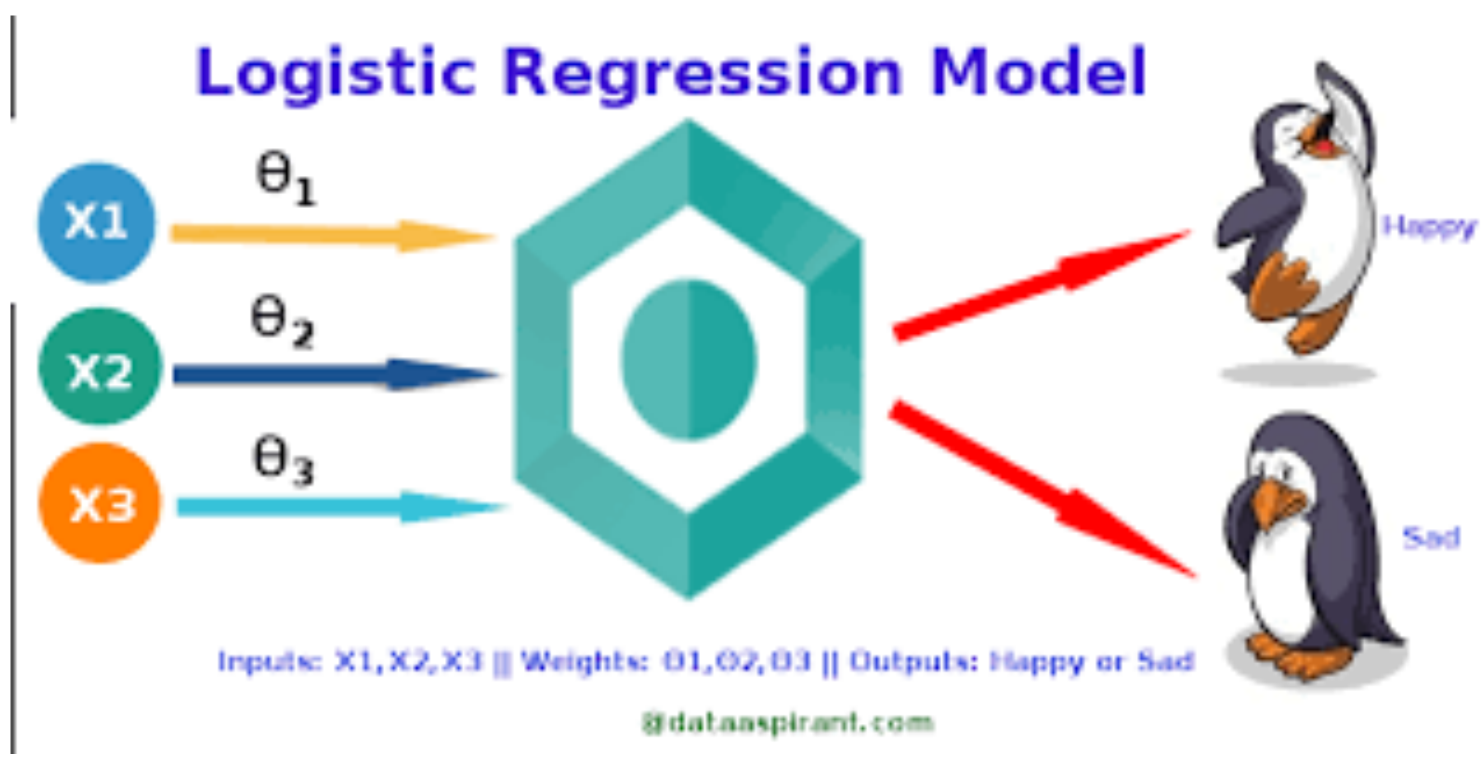
Regression Models

Logistic regression models suffer from the same problems as linear regression models.

The phenomenon of multicollinearity, which affects the significance of the regression coefficients, requires to deal with a phase of selection of explanatory variables (feature selection).

Some known characteristics of the logistic regression model are

- *accuracy usually lower than that of other classification models;*
- *greater laboriousness in the construction phase compared to other models of classification classification;*
- *extremely complex to deal with large datasets, number of explanatory variables, number of instances.*



CLASSIFICATION
SEPARATION MODELS

Separation Models

The separation models we will present are as follows:

- *Artificial Neural Networks*
- *Support Vector Machines*

Specifically regarding the *Artificial Neural Networks*, given the richness of this class of connectionist models, we will present in detail only the following classification models:

- *Feedforward Neural Networks*
- *Radial Basis Function Networks*

As far as the *Support Vector Machines* are concerned, we will present the following models:

- *Linear hard margin*
- *Linear soft margin*
- *Non-linear*

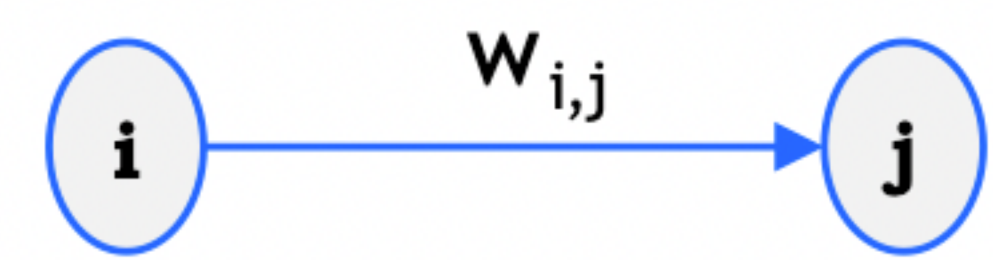
Separation Models: neural networks

Feedforward Neural Networks

Each *neuron* typically has a *set* of

- *input neurons*
- *output neurons*

Neuron *i* is *input* to neuron *j*. Neuron *j* is *output* from neuron *i*. *Neurons* are connected in an *oriented* way by the synapse that is *associated* with a real value (*weight*).



Each *neuron* is *characterized* by two elements:

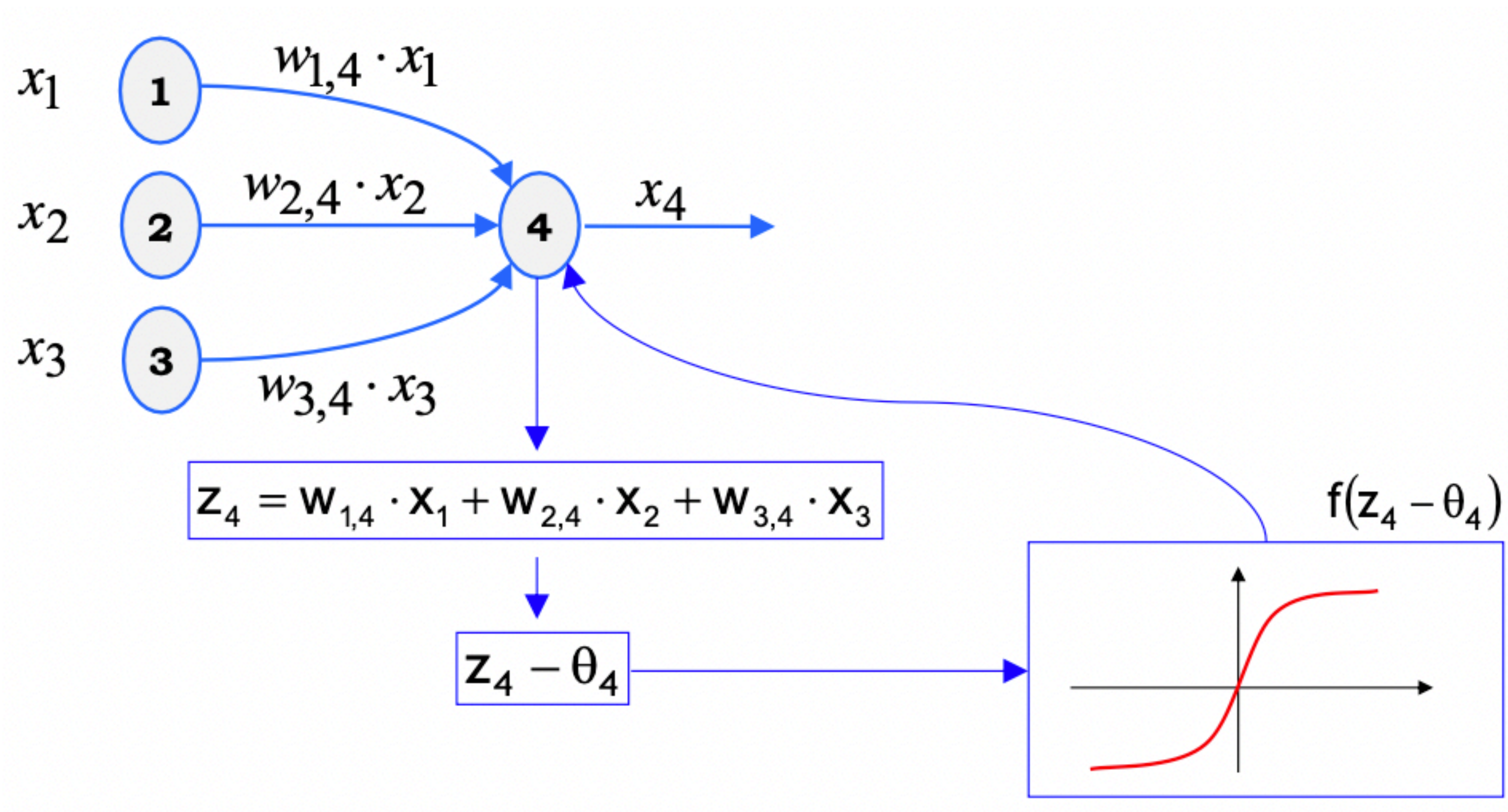
- *threshold*, bias or threshold

- *activation* or transfer *function*

Separation Models: neural networks

Each neuron:

- receives signals from other neurons (*input* neurons)
- sends signals to other neurons (*output* neurons)



Formally, neuron *j* computes the following function:

$$y_j = f\left(\sum_{i=1}^{n} w_{i,j} \cdot x_i - \theta_j\right)$$

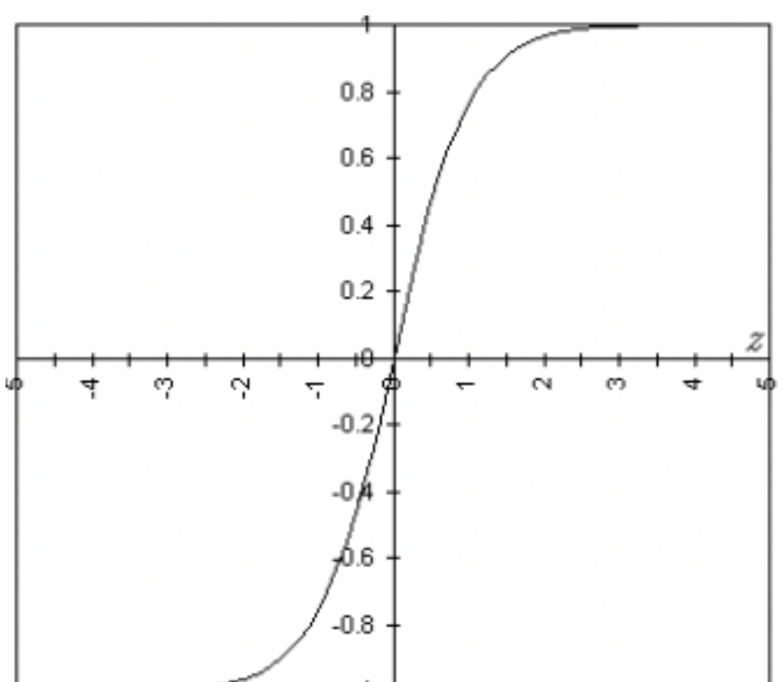
What are the *main activation functions* ?

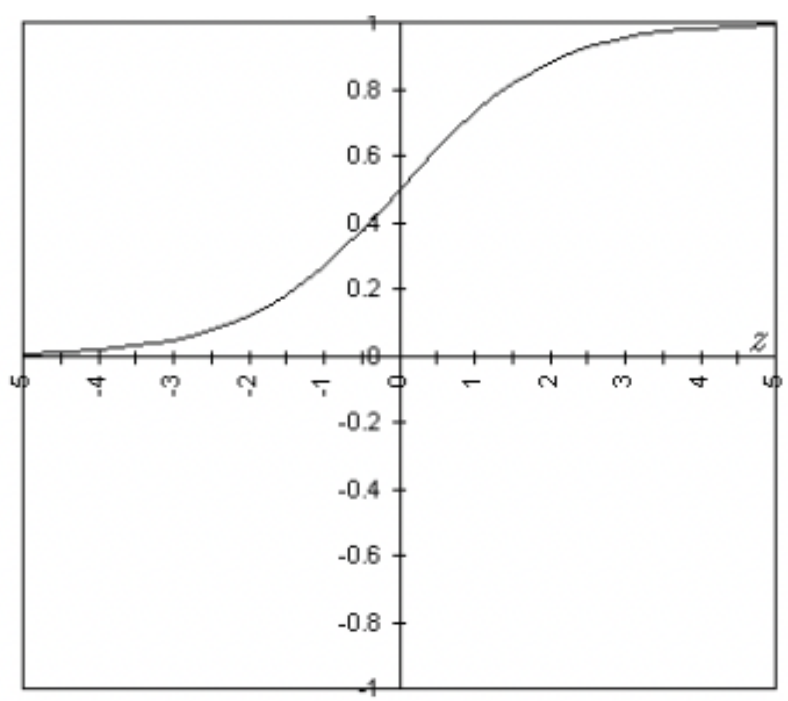
hyperbolic tangent

$$f(z) = \frac{\exp(z) - \exp(-z)}{\exp(z) + \exp(-z)}$$

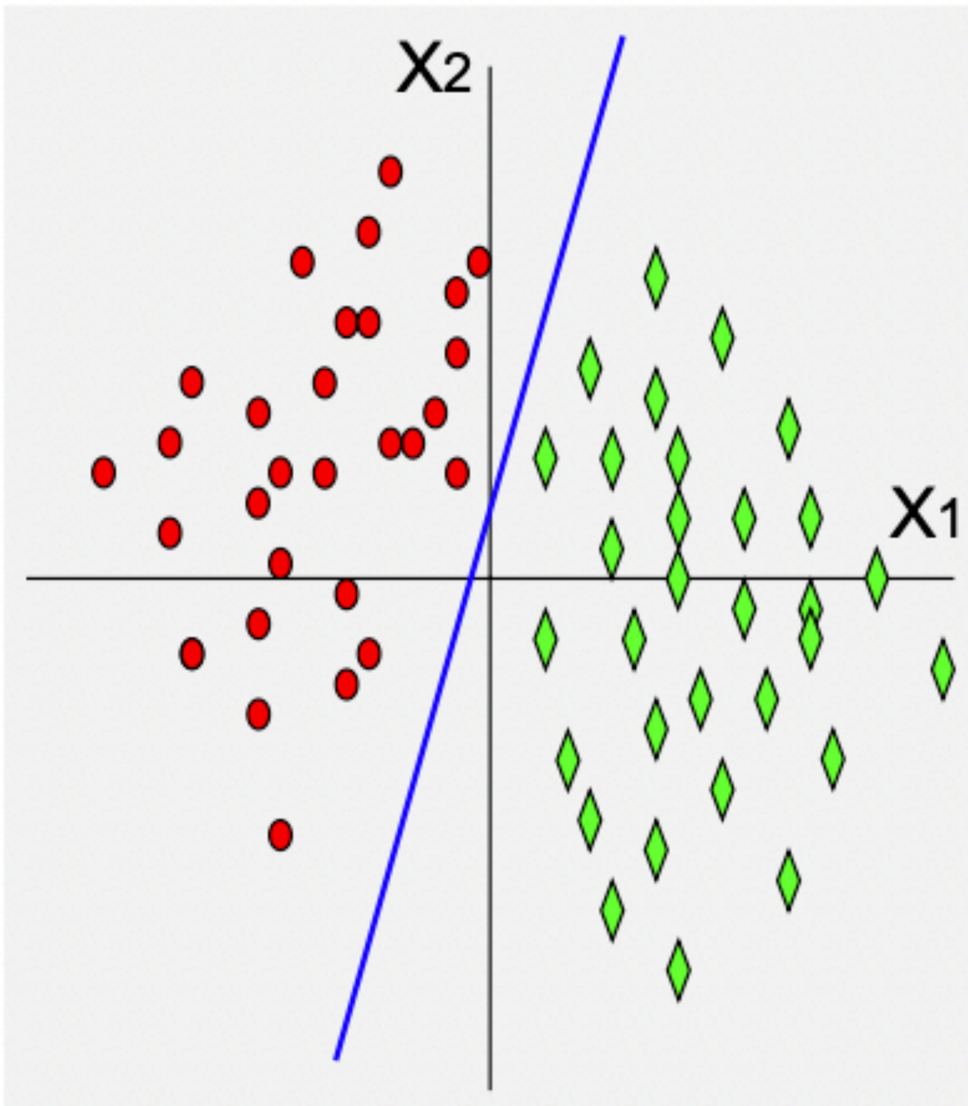
logistic

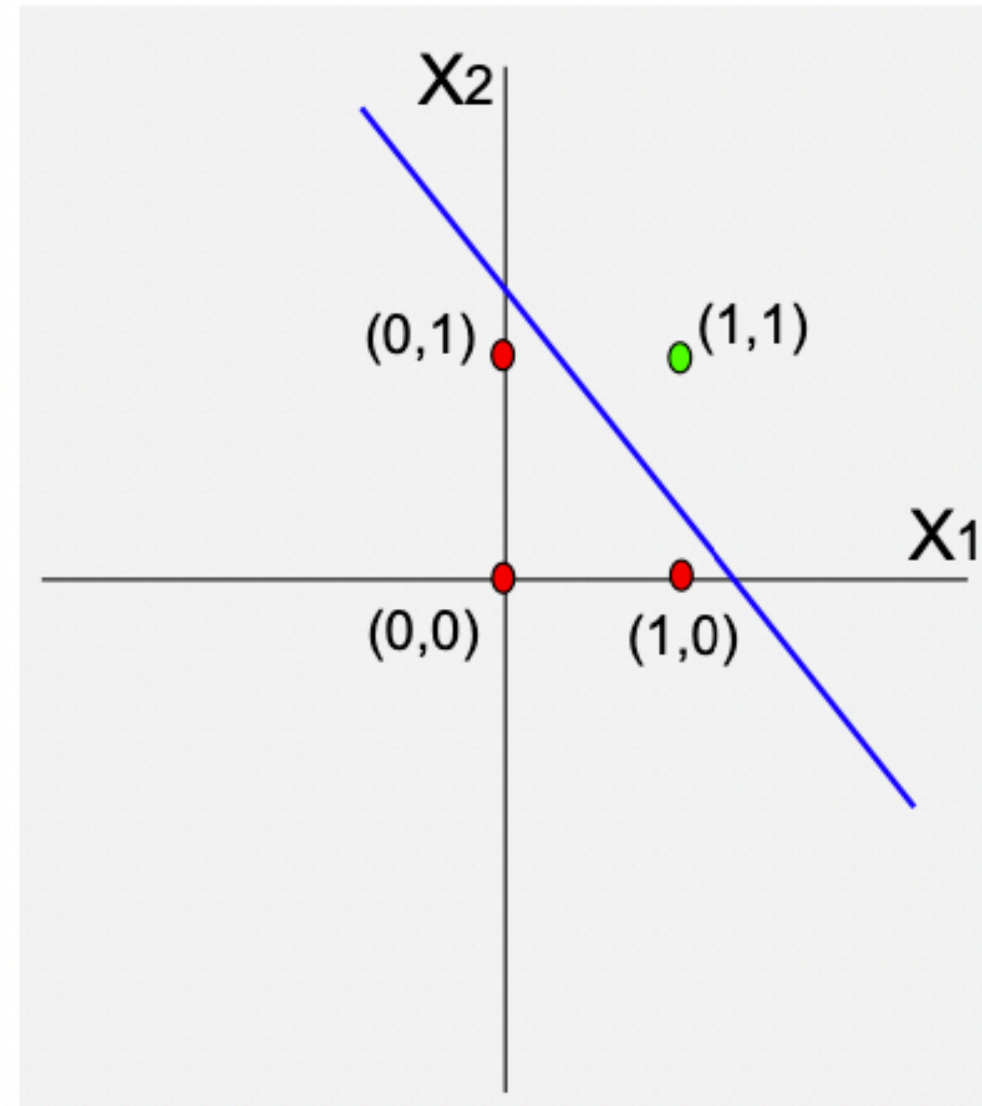
$$f(z) = \frac{1}{1 + \exp(-z)}$$





The single-layer perceptron implements a hyperplane in "*n*-dimensional" space.



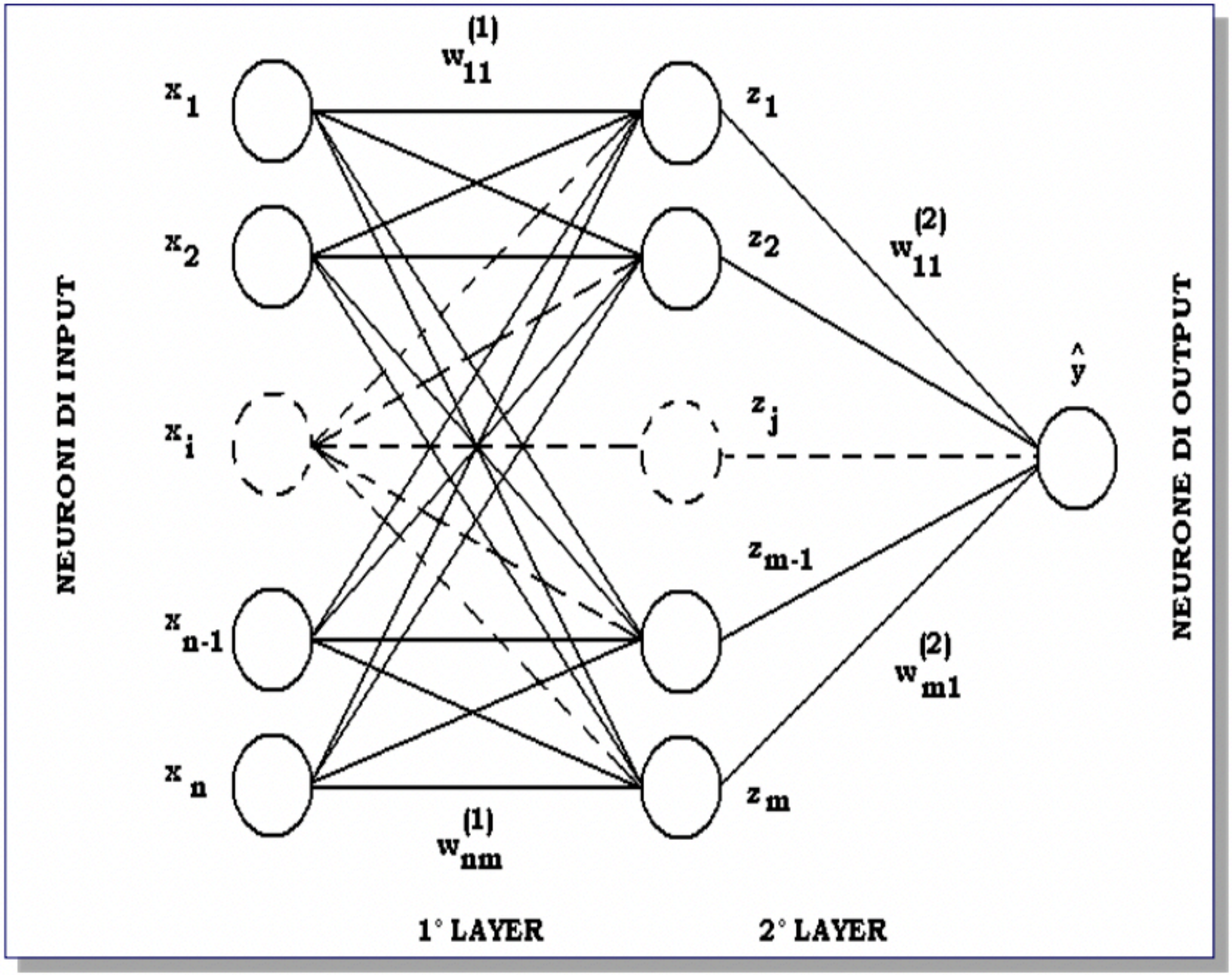


AND function

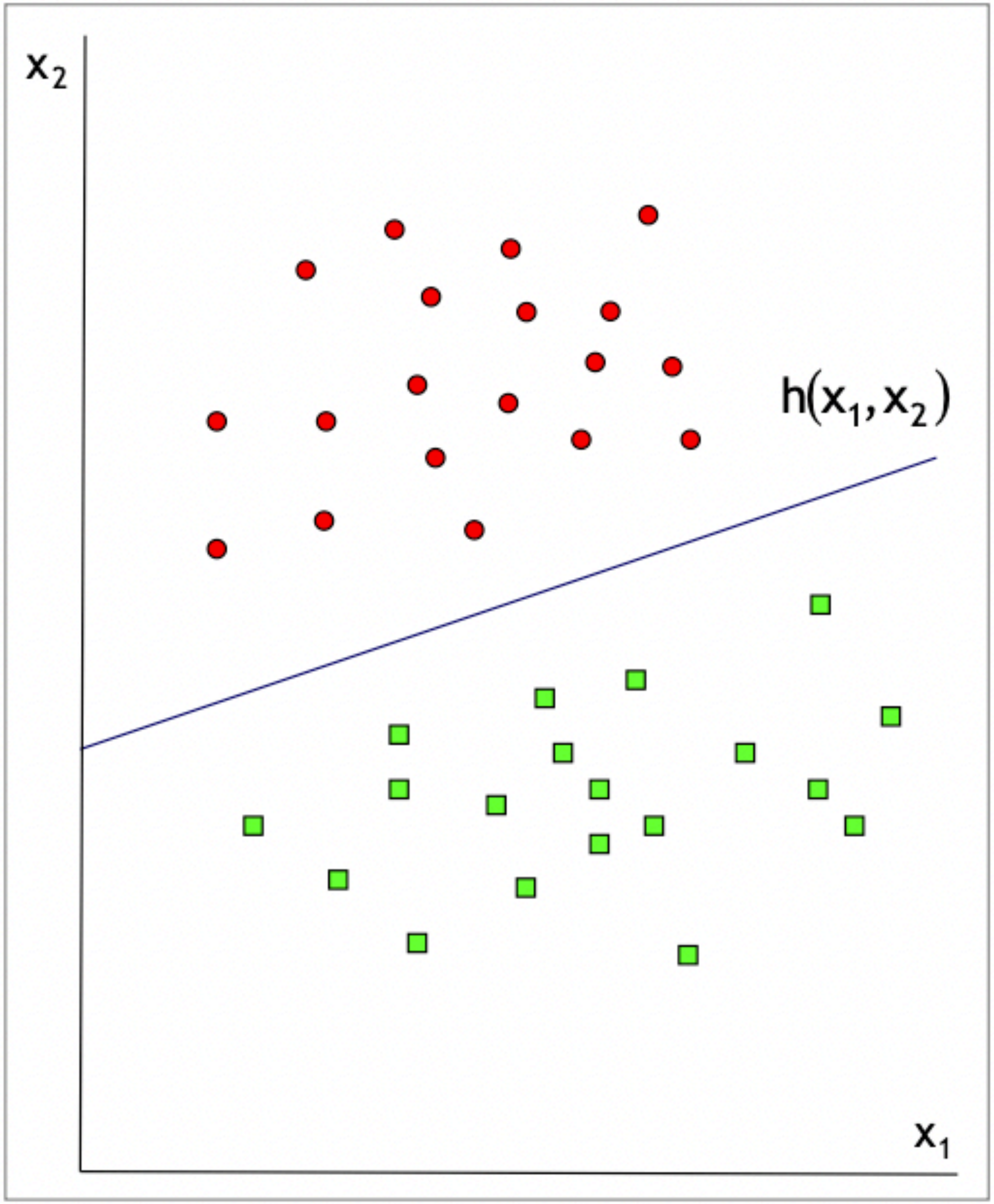
Is the *Perceptron* able to learn *any function* ?

Separation Models: neural networks

Multi-layer perceptron or Feedforward Neural Network



Separation Models: SVM



A Support Vector Machine (SVM) learns linear functions with threshold:

$$h(\underline{x}) = \mathrm{sign}\{\underline{w} \cdot \underline{x} + b\} = \begin{cases} +1 & \text{if } \underline{w} \cdot \underline{x} + b \geq 0 \\ -1 & \text{otherwise} \end{cases}$$

The function $h(\cdot)$ has as its argument a *hyperplane* in the space of explanatory variables.

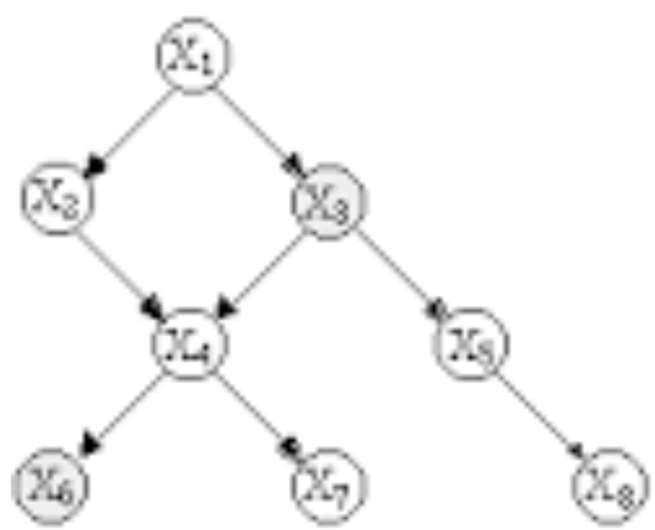
Each instance is classified according to the part of the hyperplane in which it is located.

CLASSIFICATION
HEURISTIC MODELS

Heuristic Models

They use classification procedures based on elementary and intuitive algorithmic schemes. To this category belong:

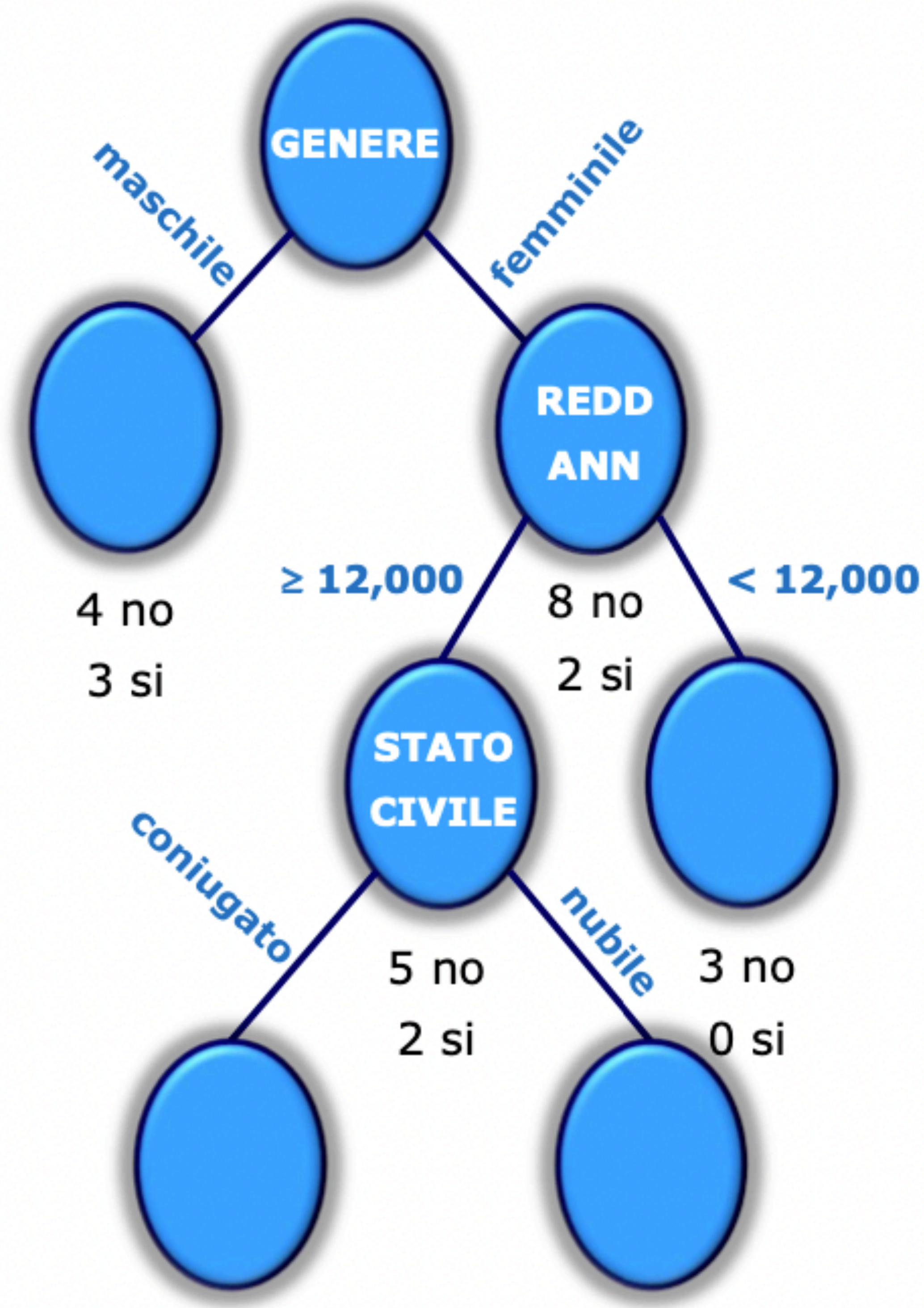
- ***Nearest Neighbor***: based on the notion of distance between observations.
- ***Classification trees***: adopt divide and conquer schemes to induce groupings of observations as homogeneous as possible with respect to the specific target class taken into account.
- ***Random Forest***: exploit the scheme of classification trees to develop efficient and effective models by combining different predictions



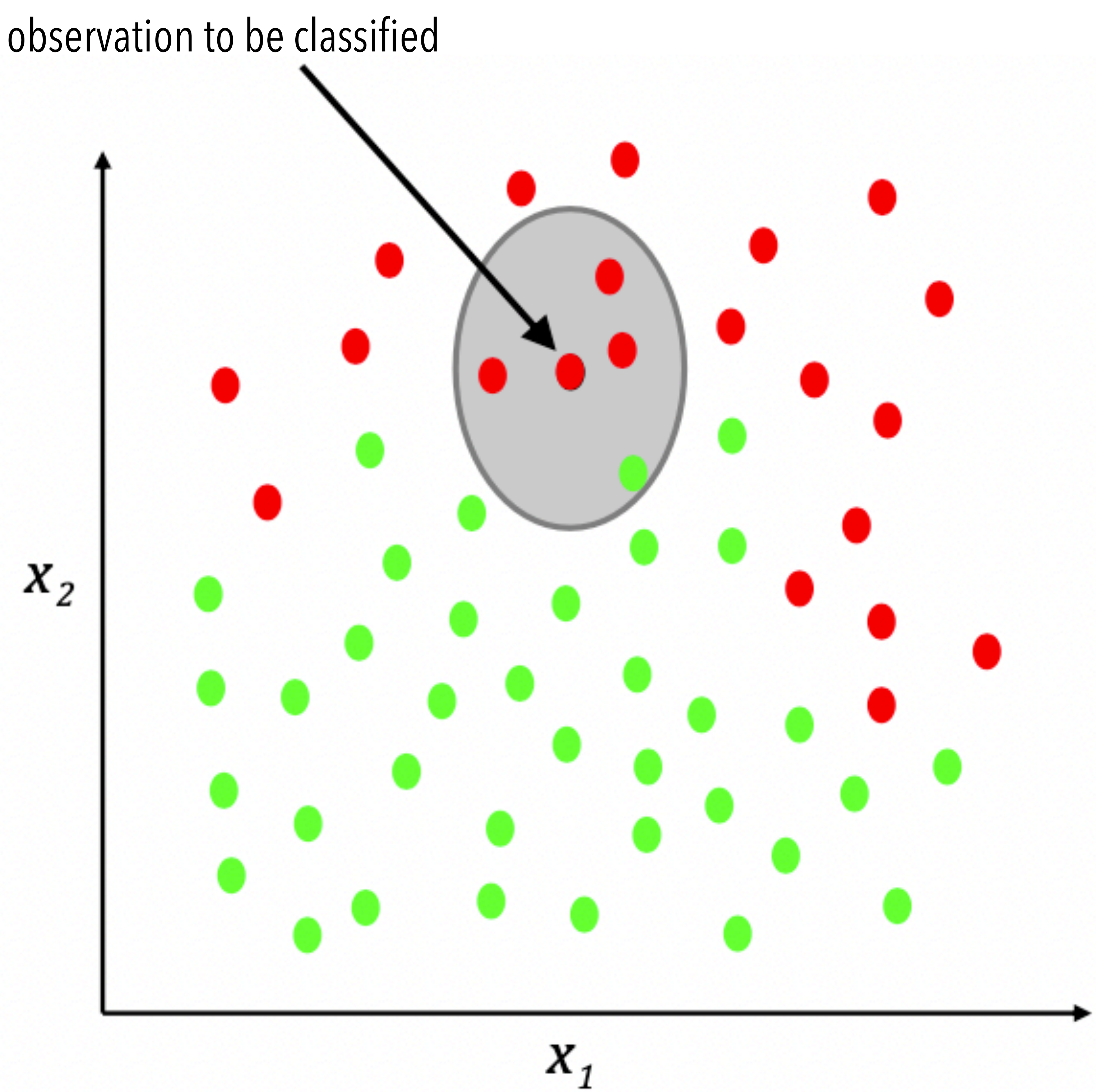
X_1	$\theta_{10} = P(X_1 = 0) = 0.2$
X_2	$\theta_{200} = P(X_2 = 0 \mid X_1 = 0) = 0.3$
	$\theta_{201} = P(X_2 = 0 \mid X_1 = 1) = 0.5$
X_3	$\boldsymbol{\theta_{300}} = P(X_3 = 0 \mid X_1 = 0)$
	$\theta_{301} = P(X_3 = 0 \mid X_1 = 1) = 0.5$
X_4	$\theta_{4000} = P(X_4 = 0 \mid X_2 = 0, X_3 = 0) = 0.1$
	$\theta_{4001} = P(X_4 = 0 \mid X_2 = 0, X_3 = 1) = 0.3$
	$\theta_{4010} = P(X_4 = 0 \mid X_2 = 1, X_3 = 0) = 0.8$
	$\theta_{4011} = P(X_4 = 0 \mid X_2 = 1, X_3 = 1) = 0.4$
X_5	$\theta_{500} = P(X_5 = 0 \mid X_3 = 0) = 0.3$
	$\theta_{501} = P(X_5 = 0 \mid X_3 = 1) = 0.1$
X_6	$\boldsymbol{\theta_{600}} = P(X_6 = 0 \mid X_4 = 0)$
	$\theta_{601} = P(X_6 = 0 \mid X_4 = 1) = 0.9$
X_7	$\theta_{700} = P(X_7 = 0 \mid X_4 = 0) = 0.3$
	$\theta_{701} = P(X_7 = 0 \mid X_4 = 1) = 0.6$
X_8	$\theta_{800} = P(X_8 = 0 \mid X_5 = 0) = 0.2$
	$\theta_{801} = P(X_8 = 0 \mid X_5 = 1) = 0.4$

Heuristic Models: classification trees

GENERE	ETA	PROVINCIA	REGIONE	REDD ANN	...	...	STATO CIVILE	EVASORE
maschile	32	VA	Lombardia	20,000 €			celibe	no
femminile	45	AQ	Abruzzo	13,500 €			coniugato	si
femminile	21	RM	Lazio	11,600 €			nubile	no
femminile	62	RM	Lazio	15,350 €			nubile	no
maschile	68	RC	Calabria	10,945 €			divorziato	no
maschile	19	AN	Marche	10,233 €			celibe	no
maschile	24	LT	Lazio	10,450 €			coniugato	si
femminile	22	VI	Veneto	11,567 €			coniugato	no
femminile	29	NO	Piemonte	16,350 €			nubile	no
maschile	52	FI	Toscana	11,245 €			nubile	si
femminile	34	MI	Sicilia	13,450 €			coniugato	no
femminile	33	MI	Basilicata	7,500 €			coniugato	no
femminile	55	TN	Trentino	13,450 €			coniugato	si
maschile	39	FR	Lazio	11,590 €			celibe	si
femminile	55	MI	Lombardia	23,500 €			coniugato	no
maschile	27	MI	Lombardia	35,800 €			celibe	no
femminile	31	AQ	Abruzzo	14,750 €			coniugato	no



Heuristic Models: **nearest neighbor**



We fix k=4

Euclidean distance

two classes

3 red, 1 green => red

You can weigh the grade of each observation inversely proportional to the distance.

CLASSIFICATION
PROBABILISTIC MODELS

Probabilistic Models

Probabilistic models solve the supervised classification problem through the use of the following conditional probability

$$P(Y \mid \underline{X})$$

where for the moment we assume that Y is a binary variable and $\underline{\boldsymbol{X}}$ is an n-dimensional binary vector.

Furthermore, we will denote by X_i the *i*-th component of the vector $\underline{\boldsymbol{X}}$.

According to *Bayes' formula* we can write

$$P(Y = y_i \mid \underline{X} = x_k) = \frac{P(\underline{X} = x_k \mid Y = y_i) \cdot P(Y = y_i)}{\sum_j P(\underline{X} = x_k \mid Y = y_j) \cdot P(Y = y_j)}$$

where y_i denotes the *i*-th element of the support of Y, while x_k denotes the *k*-th possible assignment of the vector $\underline{\boldsymbol{X}}$.