Classification: From Expert to Ensemble to Crowd

Lorenza Saitta

saitta@mfn.unipmn.it

Classification

- Classification (only indirectly learning)
- Data independent setting
- Framework where alternative classification strategies can be compared

- PROBLEMS
  - Error estimation
  - Class assignment
  - Optimization of the classification process

Parameters of the Classification Process

- Number of classifiers
  - Single
  - Ensemble (Bagging-like)
  - Crowdsourcing
- Number of classes
  - Two classes (binary)
  - More than two classes (multi-class)
- Aggregation function
  - Majority voting
  - Weighted Majority voting
  - Plurality voting

Problem Definition

- Set $\Phi = \{\varphi_j \mid 1 \leq \varphi \leq R\}$ of classifiers
- Finite set $\mathcal{X}$ of examples to classify : $|\mathcal{X}| = N$
- Examples may be duplicated – only n are different
  - Empirical frequency f_k of example x_k

$$f_k = \frac{n_k}{N} \qquad \sum_{k=1}^{n} n_k = N \qquad \sum_{k=1}^{n} f_k = 1$$

$$r_j = \sum_{k=1}^{n} f_k p_j(x_k) \qquad p_k = \frac{1}{R}\sum_{j=1}^{R} p_j(x_k) \qquad r = \frac{1}{R}\sum_{k=1}^{n}\sum_{j=1}^{R} f_k p_j(x_k)$$

Decision Matrix

	φ_1		φ_j		φ_R	p_k
x_1		…		…		p_1
x_k		…	$p_j(x_k)$	…		p_k
x_n		…		…		p_n
r_j	r_1		r_j			r

$p_j(x_k) \in \{0,1\}$
Probability that φ_j correctly classifies x_k

r_j = Probability that a given φ_j correctly classifies an *example* x_k *chosen* according to f

p_k = Probability that a given example x_k is correctly classifies by a *randomly chosen* classifier φ_j

Example of Decision Matrix

	φ_1	φ_2	φ_3	φ_4	φ_5	φ_6	φ_7	φ_8	φ_9	p_k
x_1	0	1	1	1	1	0	1	0	1	0.67
x_2	1	1	0	0	1	1	0	1	1	0.67
x_3	1	0	1	1	1	1	0	0	0	0.56
x_4	1	1	0	1	0	0	1	0	0	0.44
x_5	0	1	1	1	0	1	0	0	0	0.44
x_6	1	0	1	0	1	0	1	0	0	0.44
x_7	1	1	0	0	0	0	0	1	0	0.33
r_j	0.71	0.71	0.57	0.57	0.57	0.43	0.43	0.29	0.29	r = 0.51

Values of p_k decrease top down

Values of r_j decrease left to right

Conflicting Points of View

Classifier-centred strategy

Each classifier in Φ is only interested in how many examples in $\mathcal{X}$ it correctly classifies, independently from:

- the identity of the misclassified examples
- the accuracies of other classifiers

Example-centred strategy

Each example in $\mathcal{X}$ is only interested in being itself correctly classified, independently from:

- the identity of the used classifiers
- whether or not any other example is correctly classified

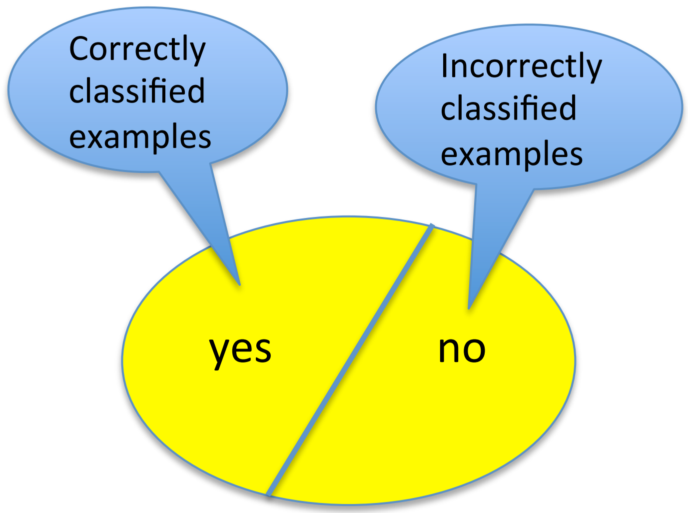
"Expert" Classification

- Classifier-centred strategy
- Monte Carlo matrix has:
  - n rows, each corresponding to a set of n_k istances $x_{k,i}$ of example x_k
  - just one column, corresponding to a single classifier φ_1
- Probability that instance $x_{k,i}$, randomly extracted from $\mathcal{X}$, is correctly classified:

$$\pi^{(s)}(x_{k,i}) = p_j(x_k) \in \{0,1\}$$

$$\mathbf{E_f}\left[\pi^{(s)}(x_{k,i})\right] = \sum_{k=1}^{n} f_k p_1(x_{k,i}) = r_1$$

$$\mathbf{Var_f}\left[\pi^{(s)}(x_{k,i})\right] = r_1\ (1 - r_1)$$



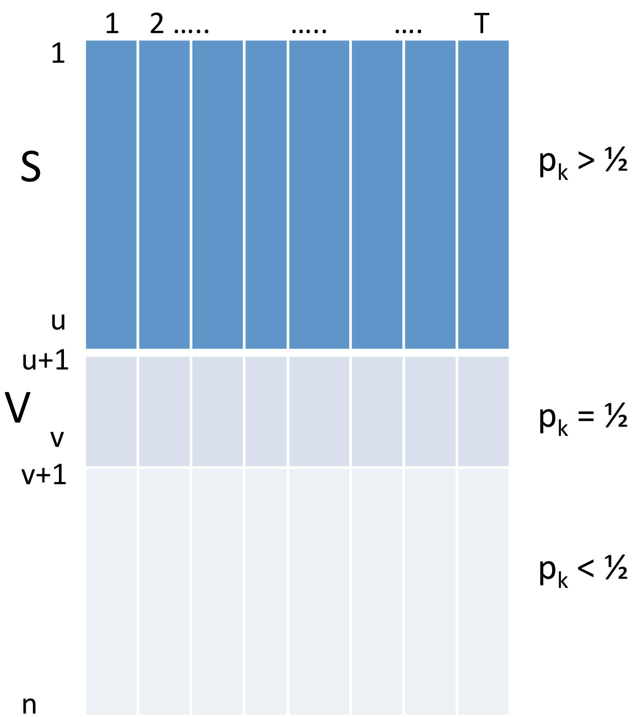
- Accuracy of the classifier (constant):

$$\rho^{(s)} = \sum_{k=1}^{n} f_k \pi^{(s)}(x_k) = r_1$$

- Assigned class: $\varphi_1(x_{k,i})$

(Bagging-like) Ensemble Classification (1)

- Classifier-centred strategy
- Monte Carlo matrix has:
  - n rows, each corresponding to a set of n_k istances $x_{k,i}$ of example x_k
  - T columns, corresponding to R classifiers $\varphi_1, \ldots, \varphi_T$
- Probability that instance $x_{k,i}$, randomly extracted from $\mathcal{X}$, is correctly classified:



$$S = \sum_{k=1}^{u} n_k \qquad \text{and} \qquad V = \sum_{k=u+1}^{v} n_k$$

$$\pi_T^{(e)}(x_{k,i}) = I_{p_k > 1/2} + \frac{1}{2}\, I_{p_k = 1/2}$$

$$\pi_T^{(e)} = \mathbf{E_f}\left[\pi^{(e)_T}(x_{k,i})\right] = \sum_{k=1}^{n} f_k\, I_{p_k > 1/2} + \frac{1}{2}\sum_{k=1}^{n} f_k\, I_{p_k = 1/2} = \frac{S}{N} + \frac{V}{2N}$$

$$\mathbf{Var_f}\left[\pi^{(e)_T}(x_{k,i})\right] = \pi_T^{(e)}\,\left(1 - \pi_T^{(e)}\right) - \frac{V}{4N}$$

$$\rho_T^{(e)} = \sum_{k=1}^{n} f_k \pi_T^{(e)}(x_{k,i}) = \sum_{k=1}^{n} f_k\, I_{p_k > 1/2} + \frac{1}{2}\sum_{k=1}^{n} f_k\, I_{p_k = 1/2} = \frac{S}{N} + \frac{V}{2N} = \pi_T^{(e)}$$

$$Max(0, 2r-1) \le \rho_T^{(e)} < Min(1, 2r)$$

(Bagging-like) Ensemble Classification (2)

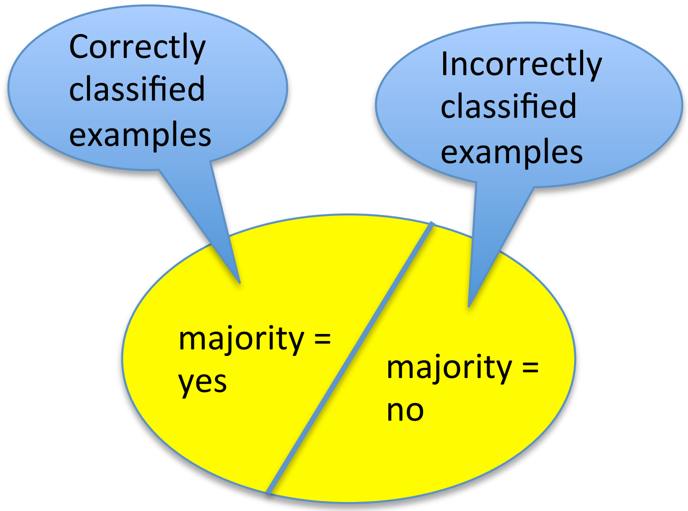
If we consider a randomly extracted instance $x_{k,i}$ from $\mathcal{X}$ according to f, we can compute the probability that it is correctly classified, independently from the other ones:

$$\pi_T^{(e)} < 1 - \sum_{s=0}^{\lfloor \frac{T}{2} \rfloor} \sum_{J_s \subseteq J_T} G(J_s) \qquad G(J_s) = \prod_{i \in J_s} r_i \prod_{i \in J_T - J_s} (1 - r_i)$$

$J_T = \{1, 2, \ldots, T\}$ $\qquad$ $J_s = \{j_1, j_2, \ldots, j_s\}$

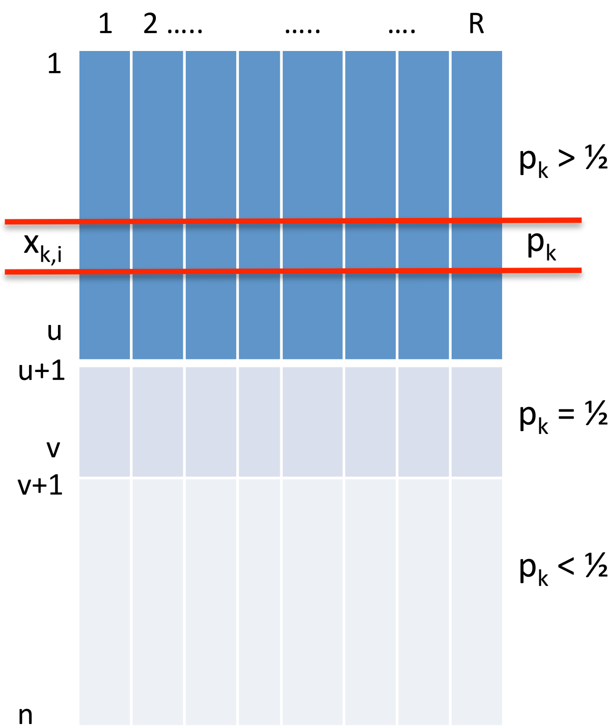
If $r_j = r \quad (1 \leq i \leq T)$:

$$\pi_T^{(e)} = 1 - \sum_{s=0}^{\lfloor \frac{T}{2} \rfloor} \binom{T}{s} r^s (1-r)^{T-s}$$



Monte Carlo Classification

- Example-centred strategy



We extract randomly, with replacement, from Φ a subset of T << R classifiers, and use them to classify $x_{k,i}$

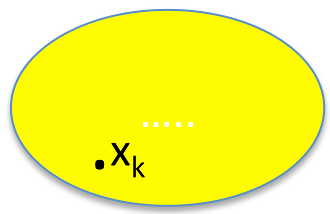
If T -> ∞, the whole set Φ is used:

$$\pi_{\infty}^{(MC)}(x_{k,i}) = I_{p_k > 1/2} + \frac{1}{2}\, I_{p_k = 1/2}$$

$$\rho_{\infty}^{(MC)} = \frac{S}{N} + \frac{V}{2N} = \rho_T^{(e)}$$

If T is finite: $$\pi_T^{(MC)}(x_{k,i}) = \sum_{t=1+\lfloor T/2 \rfloor}^{T} \binom{T}{t} p_k^t \, (1-p_k)^{T-t}$$

Each example is classified independently.
Not all examples have the same difficulty.



Mean and Variance of the Accuracy

Probability that a given subset of T classifier is chosen:

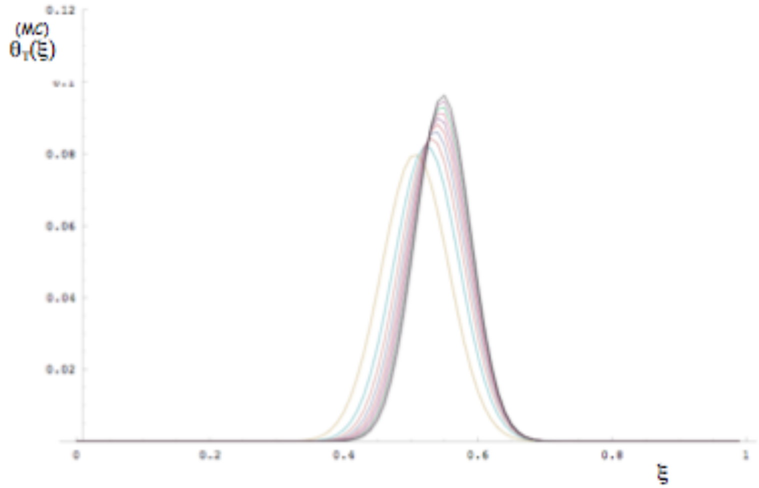
$$\zeta(\Phi_T^{(k,i)}) \stackrel{\text{def}}{=} \Pr(t_1, t_2, ..., t_R) = \binom{T}{t_1 \;\; t_2 \; ... \; t_R} \prod_{j=1}^{R} \left(\frac{1}{R}\right)^{t_j} \quad \text{with} \quad \sum_{j=1}^{R} t_j = T$$

$$\mathbf{E}_\zeta \left[\rho_T^{(MC)}\right] = \frac{1}{N}\mathbf{E}_\zeta \left[c(T)\right] = \frac{1}{N}\sum_{k=1}^{n} \mathbf{E}_\zeta \left[c_k(T)\right] = \sum_{k=1}^{n} f_k \; \pi(T, p_k)$$

$$\mathbf{Var}_\zeta \left[\rho_T^{(MC)}\right] = \frac{1}{N^2}\sum_{k=1}^{n} \mathbf{Var}_\zeta \left[c_k(T)\right] = \frac{1}{N}\sum_{k=1}^{n} f_k \; \pi(T, p_k) \; \left[1 - \pi(T, p_k)\right]$$

Probability Distribution of the Accuracy

Exact probability distribution -> classifications of examples are independent



Very good approximation:

$$\theta_T^{(MC)}(\xi) \approx \frac{1}{\sqrt{2\pi \ \mathbf{Var}[\rho_T^{(MC)}]}} \, e^{-\frac{\left(\xi - \mathbf{E}\left[\rho_T^{(MC)}\right]\right)^2}{2 \ \mathbf{Var}\left[\rho_T^{(MC)}\right]}}$$

Crowdsourcing Classification

- Example-centred strategy
- Number R of *workers* (classifiers) and number N of *tasks* (examples) very large
- Each example is classified by T workers and the answers are aggregated
- Reliability of workers is unknown (*hammer*, *spammer*)

Crowdsourcing classification can be performed EXACLTY as Monte Carlo suggests

Crowdsourcing and Machine Learning (ML)

- Single classification is impossible
- Ensemble is impractical, especially Adaboost-like (we do not know the true labels of examples)
- ML is not concerned with the way in which classification is performed
- Monte Carlo theory allows redundancies and independencies to be exploited at their best
- Multi-class problem: impossible to reduce to pairwise classification or one-vs-all

Monte Carlo classification handles in a uniform way both binary and multiclass problems

Label Assignment

Most difficult problem

- Accuracy of workers is unknown
- Ground truth is unavailable

An extra criterion is needed

- Aggregation
- Hypothesis about the distribution of the workers among hammers and spammers
- Correct classification satisfies Maximum Likelihood principle for some parameter

Two approaches

- One-step aggregation
- Iterative process (EM alternating computation of probability of label and reliability of workers)

Summary

- Monte Carlo framework
  - Binary/multiclass
  - Spammer resistant
  - Accuracies of workers not required to be all greater than ½

- Experiments
  - (Amazon Mechanical Turk)
  - Hung et al (2013) -> Framework for synthetic data

Conclusions

- Trade-off between Quantity and Quality
- Concentrated vs diffuse knowledge
- Generating knowledge has a cost
- Human -> Machine -> Human

- Crowdsourcing is well suited to problems where the natural abilities of humans can be easily exploited

- Relationships with Machine Learning?