COST-SENSITIVE LEARNING: IN A NUTSHELL

- Cost-sensitive learning:
 - Classical learning: data sets are balanced, and all errors have equal costs
 - We now assume given, unequal cost
 - And try to minimize them in expectation
- Applications:
 - Medicine Misdiagnosing as healthy vs. having a disease
 - (Extreme) Weather prediction Incorrectly predicting that no hurricane occurs
 - Credit granting Lending to a risky client vs. not lending to a trustworthy client.

		Truth		
		Default	Pays Back	
Pred.	Default	0	10	
	Pays Back	1000	0	

- In these examples, the costs of a false negative is much higher than the costs of a false positive.
- In some applications, the costs are unknown ~→ need to be specified by experts, or be learnt.



COST MATRIX

• Input: cost matrix C

		True Class y				
		1	2		g	
Classification	1	C(1, 1)	C(1, 2)		C(1, g)	
	2	C(2, 1)	C(2, 2)		C(2, g)	
Ŷ						
	•					
	•		- · · · · ·		- · · ·	
	g	C(g, 1)	C(g, 2)		C(g,g)	



- C(j, k) is the cost of classifying class k as j,
- 0-1-loss would simply be: $C(j, k) = \mathbb{1}_{[j \neq k]}$
- C designed by experts with domain knowledge
 - Too low costs: not enough change in model, still costly errors
 - Too high costs: might never predict costly classes

COST MATRIX FOR IMBALANCED LEARNING

- Common heuristic for imbalanced data sets:
 - C(j, k) = n_j/n_k with n_k ≪ n_j, misclassifying a minority class k as a majority class j
 - C(j,k) = 1 with $n_j \ll n_k$,

misclassifying a majority class k as a minority class j

• 0 for a correct classification

• Imbalanced binary classification:

True classy = 1y = -1Pred. $\hat{y} = 1$ 0 $\hat{y} = -1$ $\frac{n_-}{n_+}$ 0

• So: much higher costs for FNs

× × 0 × × ×

MINIMUM EXPECTED COST PRINCIPLE

- Suppose we have:
 - a cost matrix C
 - knowledge of the true posterior $p(\cdot | \mathbf{x})$
- Predict class j with smallest expected costs when marginalizing over true classes:

$$\mathbb{E}_{K \sim p(\cdot \mid \mathbf{x})}(C(j, K)) = \sum_{k=1}^{g} p(k \mid \mathbf{x})C(j, k)$$

• If we trust we trust a probabilistic classifier, we can convert its scores to labels:

$$h(\mathbf{x}) := \operatorname*{arg\,min}_{j=1,\ldots,g} \sum_{k=1}^{g} \pi_k(\mathbf{x}) C(j,k).$$

• Can be better to take a less probable class (• Elkan et. al. 2001)





OPTIMAL THRESHOLD FOR BINARY CASE

- Optimal decisions do not change if
 - C is multiplied by positive constant
 - C is added with constant shift
- Scale and shift C to get simpler C':

True class
$$y = 1$$
 $y = -1$ Pred. $\hat{y} = 1$ $C'(1,1)$ 1class $\hat{y} = -1$ $C'(-1,1)$ 0

× × 0 × × ×

where

•
$$C'(-1,1) = \frac{C(-1,1)-C(-1,-1)}{C(1,-1)-C(-1,-1)}$$

• $C'(1,1) = \frac{C(1,1)-C(-1,-1)}{C(1,-1)-C(-1,-1)}$

• We predict **x** as class 1 if

$$\mathbb{E}_{\mathcal{K} \sim \rho(\cdot \mid \mathbf{x})}(\mathcal{C}'(1, \mathcal{K})) \leq \mathbb{E}_{\mathcal{K} \sim \rho(\cdot \mid \mathbf{x})}(\mathcal{C}'(-1, \mathcal{K}))$$

OPTIMAL THRESHOLD FOR BINARY CASE / 2

• Let's unroll the expected value and use C':

$$p(-1 \mid \mathbf{x})C'(1, -1) + p(1 \mid \mathbf{x})C'(1, 1) \le p(-1 \mid \mathbf{x})C'(-1, -1) + p(1 \mid \mathbf{x})C'(-1, 1)$$

$$\Rightarrow [1 - p(1 \mid \mathbf{x})] \cdot 1 + p(1 \mid \mathbf{x})C'(1, 1) \le p(1 \mid \mathbf{x})C'(-1, 1)$$

$$\Rightarrow p(1 \mid \mathbf{x}) \ge \frac{1}{C'(-1, 1) - C'(1, 1) + 1}$$

$$\Rightarrow p(1 \mid \mathbf{x}) \ge \frac{C(1, -1) - C(-1, -1)}{C(-1, 1) - C(1, 1) + C(1, -1) - C(-1, -1)} = c^*$$

• If even C(1, 1) = C(-1, -1) = 0, we get:

$$p(1 \mid \mathbf{x}) \geq \frac{C(1,-1)}{C(-1,1) + C(1,-1)} = c^*$$

• Optimal threshold *c*^{*} for probabilistic classifier

$$h(\mathbf{x}) := 2 \cdot \mathbb{1}_{[\pi(\mathbf{x}) \ge c^*]} - 1$$