Decision Support Systems

Utility in ML

Andrea Campagner, PhD, Post-Doc Researcher

IRCCS Istituto Ortopedico Galeazzi, Milan, Italy
MUDI Lab, Department of Computer Science, University of Milano-Bicocca, Milan, Italy

Contacts: onyris93@gmail.com; andrea.campagner@unimib.it
https://andreacampagner.github.io/

18 April 2023

Table of Contents

Brief Recap on ML

Let X be a feature space and Y be a target space. A *data generating distribution* is a probability distribution $\mathcal{D}$ over $X \times Y$.

Sometimes we assume that $\mathcal{D}$ is *deterministic*, i.e. exists $f : X \to Y$ with $\mathcal{D}(y|x) = 1$ iff $f(x) = y$.

Let $\mathcal{H}$ be a class of models (i.e., functions $X \to Y$), and $l : Y \times Y \to \mathbb{R}$ be a *loss function*: intuitively, $l(y', y)$ is the cost of saying y' when the correct answer would be y.

The goal of ML is to design an algorithm A that allows us to select a model h for each finite sample $S \sim \mathcal{D}^m$ s.t. $\mathbb{E}_{(x,y)\sim\mathcal{D}} l(h(x), y)$ is *small*.

Utilities and Loss Functions

Loss functions inherently describe a utility function (or, dually, a regret function) for our learning problem.

For example, the 0-1 loss associates utility 1 (regret 0) when $y = y'$ and otherwise utility 0 (regret 1).

Predicted \ True	Positive class	Negative class
Positive class	1	0
Negative class	0	1

Applications of Loss Functions

Crucially, loss functions (dually, utilities) have two applications: *training* and *evaluation*

In training, we use the loss function as the objective of an optimization problem (*empirical risk minimization*): $\arg\min_{h \in \mathcal{H}} \frac{1}{|Tr|} \sum_{(x,y) \in Tr} l(h(x), y)$

In evaluation, we use the loss function to estimate the real (population-wise) loss of a given selected model: $\frac{1}{|Te|} \sum_{(x,y) \in Te} l(h(x), y)$

In this lecture we focus on evaluation!

Cost-sensitive Classification

Most common metric for evaluation of ML models is *accuracy*:

$$\frac{|\{(x, y) \in Te : h(x) = y\}|}{|Te|}$$

Accuracy corresponds to the 0-1 loss: assumes that all decisions (both errors and correct ones) have the same cost/utility... hardly true in practice!

Predicted \ True	Cancer	Not Cancer
Cancer	Patient improves	Side-effects
Not Cancer	Patient Dies	OK

Cost-sensitive Classification (cont.)

Cost-sensitive Classification aims to weight different decisions differently!

Predicted \ True	Positive	Negative
Positive	a	b
Negative	c	d

$$Cost(h, Te) = a|TP| + b|FP| + c|FN| + d|TN|$$

The costs should reflect some notion of utility in the considered application!

Limitations of Cost-sensitive Classification

Cost-sensitive classification considers models as *classification supports* (functions $X \to Y$) and the target as deterministic... However:

- Our models are *probabilistic supports*: functions $X \to \Delta(Y)$ that associate with each x a probability distribution over y
- The data generating distribution $\mathcal{D}$ is not deterministic (e.g., what happens when some relevant features are not observed?)

In these settings, (cost-sensitive) accuracy is not enough and we want other guarantees (e.g., *calibration*: $h(x)$ should be *close* to $\mathcal{D}(\cdot|x)$)

Net Benefit and Decision Curves

Net benefit theory aims to address the limitations of simple cost-sensitive classification in a principled manner.

It decouples prediction from decision:

- **Prediction**: Our models are probabilistic $X \to [0, 1]$, where $h(x)$ denotes the probability of the positive class;
- **Decision**: We get a label $\hat{y} \in \{0, 1\}$ by thresholding $h(x)$ at a fixed threshold τ

Net Benefit and Decision Curves (cont.)

Net benefit theory gives two main tools:

- A way to select a threshold and evaluate models at this value (**net benefit metric**);
- A qualitative way to evaluate a model at different decision thresholds (**decision curves**);

Net Benefit: Selecting a Threshold

In the cost-sensitive setting, imagine the threshold τ at which we would be most undecided between the two labels:

$$\tau a + (1-\tau)b = \tau c + (1-\tau)d \implies$$

$$\frac{1-\tau}{\tau} = \frac{a-c}{d-b} \implies$$

$$\tau = \frac{\text{Cost}}{\text{Cost} + \text{Benefit}}$$

where $\text{Cost} = d - b$ is the cost of a false positive, $\text{Benefit} = a - c$ is the utility of a true positive: you should set the decision threshold based on your utilities!

Net Benefit: Deriving a Metric

Assume we fix Benefit $= a - c = 1$, then the value of a false positive is:

$$b - d = -\frac{\tau}{1-\tau}$$

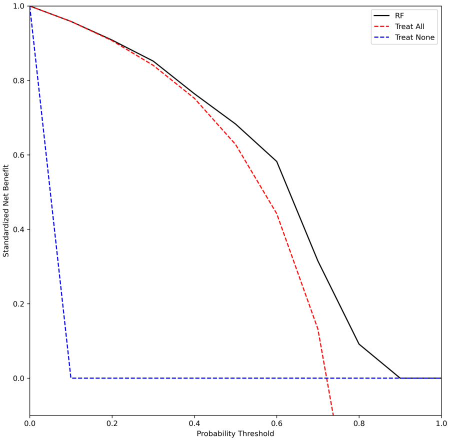
Then the utility (**net benefit**) of our model, at threshold τ is defined as:

$$NB(\tau) = \frac{|TP|}{n} - \frac{\tau}{1-\tau}\frac{|FP|}{n}$$

Intuitively, the net benefit evaluates our model by checking how many times we correctly get the positive cases, discounted by the fraction of false positives: **however, false positives are weighted by their cost**!

Net Benefit: Decision Curves

What if we do not precisely know the utilities? We can compute the net benefit at different thresholds and get a **decision curve**



Net Benefit: Variations and Current Research

- NB has range $[-\infty, \pi]$ where π is the proportion of positive labels... if we multiply the *FP* term by $\frac{1-\pi}{\pi}$ we obtain the **Standardized Net Benefit**, with range $[-\infty, 1]$;
- One can consider different decision thresholds for different instances: **Weighted Utility** [1];
- Net Benefit can be related to other metrics, e.g. AUC and ROC curves [2]

1. Campagner, A., Sternini, F., Cabitza, F. (2022). Decisions are not all equal—Introducing a utility metric based on case-wise raters' perceptions. Computer Methods and Programs in Biomedicine, 221, 106930.
2. Carrington, A. M., Fieguth, P. W., Mayr, et al. (2022). The ROC Diagonal is Not Layperson's Chance: A New Baseline Shows the Useful Area. In: International Cross-Domain Conference, CD-MAKE 2022.