1 Risk

1.1 Recap

In lecture, we generalized our notion of loss from basic 0/1 loss to incorporating the idea of risk:

Risk: $R(h) = \mathbb{E}_{XY}[L(Y, h(x)]$

For two class 0/1 loss we then saw our optimal classification function as:

$h^*(x) = \arg\max_{y \in \{0,1\}} P(Y = y|X = x)$

1.2 More General Loss

We will now work through a problem in which we have two-classes, but different loss.

We want to design an automated fishing system that captures fish, classifies them, and sends them off to two different companies, Goldfish, Inc., and Blue Tang, Inc. For some reason we only ever catch goldfish and blue tang. Goldfish, Inc. wants goldfish, and Blue Tang, Inc. wants blue tang. Given only the weights of the fish we catch, we want to figure out what type of fish it is using machine learning! Let us assume that the weight of both goldfish and blue tang are both normally distributed (univariate Gaussian), given by the p.d.f.:

$$P(x|\mu_i, \sigma_i) = \frac{1}{\sigma_i\sqrt{2\pi}} e^{-\frac{(x-\mu_i)^2}{2\sigma_i^2}}$$

We are given this data:

Data for goldfish: $\{3, 4, 5, 6, 7\}$

Data for blue tang: $\{5, 6, 7, 8, 9, 7 + \sqrt{2}, 7 - \sqrt{2}\}$

When we classify blue tang incorrectly, it gets sent to Goldfish, Inc. who won't pay us for the wrong fish and sells it themselves. When we classify goldfish incorrectly, it gets sent to Blue Tang, Inc., who is nice and returns our fish. This situation gives rise to this loss matrix:

	goldfish	blue tang
goldfish	0	100
blue tang	10	0

where the rows represent the predicted, and the columns represent the truth.

1.2.1 MLE

We give you the following maximum likelihood estimates for both classes:

Goldfish(class 1):
$\mu_1 = 5$
$\sigma_1 = \sqrt{2}$

$\pi_1 = \frac{5}{12}$

Blue Tang (class 2):

$\mu_2 = 7$
$\sigma_2 = \sqrt{2}$
$\pi_2 = \frac{7}{12}$

1.2.2 0/1 Loss

Next, find the decision rule when assuming a 0-1 loss function. Recall that a decision rule for the 0-1 loss function will minimize the probability of error.

1.2.3 Different Loss

Now, find the decision rule using the loss matrix above. Recall that a decision rule, in general, minimizes the risk, or expected loss. Let L_{ij} denote the loss for the i,j entry in the loss matrix.

2 PAC Learning

2.1 Some Important Definitions and Theorems

1. Basic notations:

- **True function** (expert/oracle) $c^* : X \to Y$ (unknown)
- Hypothesis space $\mathcal{H}$ and hypothesis $h \in \mathcal{H} : X \to Y$
- Probability Distribution p^* (unknown)
- Training Dataset $S = x^{(1)}, ...x^{(N)}$

2.True Error (expected risk)

$$R(h) = P_{x \sim p^*(x)}(c^*(x) \neq h(x))$$

3.Train Error (empirical risk)

$$\hat{R}(h) = P_{x \sim S}(c^*(x) \neq h(x)) = \frac{1}{N} \sum_{i=1}^{N} = \mathbb{1}(c^*(x^{(i)}) \neq h(x^{(i)})) = \frac{1}{N} \sum_{i=1}^{N} \mathbb{1}(y^{(i)}) \neq h(x^{(i)}))$$

4. **PAC criterion** is that we produce a high accuracy hypothesis with high probability. More formally,

$$P(\forall h \in \mathcal{H}, |R(h) - \hat{R}(h)| \leq \varepsilon) \geq 1 - \delta$$

5. A hypothesis $h \in \mathcal{H}$ is **consistent** with training data S if $\hat{R}(h) = 0$ (zero training error/correctly classify)

6. **Sample Complexity** is the minimum number of training examples N such that PAC criterion is satisfied for a given ε (arbitrarily small error) and δ (with high probability)

7. Agnostic and Realizable:

- **Realizable** means $c^* \in \mathcal{H}$
- **Agnostic** means c^* may or may not be in $\mathcal{H}$

Theorem 1. *Finite $\mathcal{H}$ and realizable case: $N \geq \frac{1}{\varepsilon}[\log|\mathcal{H}| + \log\frac{1}{\delta}]$ labelled examples are sufficient so that with probability $1 - \delta$, **all** $h \in \mathcal{H}$ with $\hat{R}(h) = 0$ have $R(h) \leq \varepsilon$.*

Proof Sketch:

- Assume k bad hypotheses $h_1, h_2, ..., h_k$ with $R(h) > \varepsilon$
- Consider a single bad hypothesis h_i. The probability it is consistent with first N data points is $\leq (1-\varepsilon)^N$ (**Note: I.I.D. assumption important here**)
- Using union bound and since $k \leq |H|$, the probability of at least one bad hypothesis is $\leq |H|(1-\varepsilon)^N$
- Fact: $1 - x \leq e^{-x}$
- The probability of at least one bad hypothesis is $\leq |H|e^{-\varepsilon N}$
- Final Step: Calculate N such that $|H|e^{-\varepsilon N} \leq \delta$

2.2 PAC-Learnable?

Consider the following problem:

Suppose we want to learn any arbitrary boolean function on M variables, i.e. $\mathcal{H} = \{f : \{0,1\}^M \to \{0,1\}\}$.

If $M = 10$, $\varepsilon = 0.01$, $\delta = 0.001$, how many examples suffice according to Theorem 1?