Currently covered

- [x] Chapter 2
- [ ] Chapter 3

Chapter 2: A gentle start

Terminology:

- Domain set (instance space): $\mathcal{X}$, all the objects (instances) we may wish to label. Represented as a vector of features.
- Label set: $\mathcal{Y}$, set of all possible labels, generated by some unknown *true* labelling function f
- Training data $(x, y) \in D, \quad |D| = m$

Empirical risk minimization

The *true error* is defined over the data generating probability distribution $\mathcal{D}$ and labeling function f

$$L_{\mathcal{D},f}(h) \stackrel{def}{=} \mathcal{P}_{x\sim\mathcal{D}}[h(x) \neq f(x)] \stackrel{def}{=} \mathcal{D}(\{x : h(x) \neq f(x)\})$$

The *training error* is defined over the training set (sample) S

$$L_S(h) \stackrel{def}{=} \frac{|\{i \in [m] : h(x_i) \neq y_i\}|}{m}$$

Overfitting: a trivial classifier that achieves zero training error simply copies labels from examples in the training set, and outputs a random label if the input vector is not from the training set.

A **Hypothesis class** is a set of predictors (family of models) $\mathcal{H}$. Each $h \in \mathcal{H}$ is a function mapping $\mathcal{X}$ to $\mathcal{Y}$.

Formally, the predictor chosen by the ERM rule is the one that minimizes the training error.

$$ERM_{\mathcal{H}}(S) \in \arg\min_{h\in\mathcal{H}} L_S(h) \tag{1}$$

The choice to restrict the hypothesis space is called an *inductive bias*. The choice of a family of predictors should be based on some prior knowledge about the problem. Ideally, we would want guarantees that the chosen family will not overfit.

DEFINITION 2.1 (The realizability assumption): There exists $h^* \in \mathcal{H}$ s.t. $L_{(\mathcal{D},f)}(h^*) = 0$

Note: this definition holds for any S – S can be any random sample from the true data distribution D labelled by f.

IID assumption All samples are *independently* and *identically* distributed according to $\mathcal{D}$: $S \sim \mathcal{D}^m$.

Two new parameters are introduced, the *confidence* of the distribution sample and *accuracy* of the classifier.

- **Confidence**: parameter δ denotes the probability of getting a **nonrepresentative** sample of the true distribution. Thus, we call $(1 - \delta)$ the *confidence parameter*
- **Accuracy**: the accuracy parameter ϵ determines what we consider as failure of the classifier. If $L_{(\mathcal{D},f)}(h_s) > \epsilon$ we consider this a failure of the learner, while $L_{(\mathcal{D},f)}(h_s) \leq \epsilon$ we consider the algorithm an *approximately correct* predictor.

With these parameters, we can formally express the *probability* to sample an $m-$tuple of instances that will lead to failure of teh learner. We define the *bad* hypotheses as the subset of the hypothesis space which has error greater than ϵ on the *true* distribution.

$$\mathcal{H}_b = \{h \in \mathcal{H} : L_{(D,f)}(h) > \epsilon\}$$

Also, we define the *miseleading* samples as the data samples which allow a hypothesis from the set of *bad* hypotheses to minimize the training error (obtain error of 0).

$$M = \{S_x : \exists h \in \mathcal{H}_B s.t. L_S(h) = 0\} \tag{2}$$

We want to upper bound the number of training samples which produce *bad* hypotheses (which obtain error lager then ϵ) as the minimizers of the *training* error.

$$\mathcal{D}^m(\{S_x : L_{(\mathcal{D},f)}(h_S) > \epsilon\})$$

We can rewrite (2) as

$$M = \bigcup_{h \in \mathcal{H}_B} \{S_x : L_s(h) = 0\}$$

the number of failures is bounded by the number of misleading samples (the *less than* comes from (1) due to the ERM being one out of multiple possible minimizers)

$$\mathcal{D}^m(\{S_x : L_{(\mathcal{D},f)}(h_S) > \epsilon\}) \leq \mathcal{D}^m(M) = \mathcal{D}^m(\cup_{h \in \mathcal{H}_B}\{S_x : L_s(h) = 0\}) \quad (3)$$

Union bound: for two sets A, B and a distribution $\mathcal{D}$, the union bound states that the size of the union of two sets is at most the sum of the sizes of both set (holds with equality when both sets are disjoint):

$$\mathcal{D}(A \cup B) \leq \mathcal{D}(A) + \mathcal{D}(B)$$

By union bounding the RHS of (3), we obtain

$$\mathcal{D}^m(\{S_x : L_{(\mathcal{D},f)}(h_S) > \epsilon\}) \leq \sum_{h \in \mathcal{H}_B} \mathcal{D}^m(\{S_x : L_s(h) = 0\})$$

TODO: annotate remaining math steps to corollary

COROLLARY 2.3 $\mathcal{H}$ is a finite hypothesis class. Let $\delta \in (0,1)$, $\epsilon > 0$ and m is an integer satisfying

$$m \geq \frac{log(|\mathcal{H}|/\delta)}{\epsilon}$$

Then, for **any** labeling function f and **any** distribution $\mathcal{D}$ for which the realizability assumption holds, with probability of *at least* $1 - \delta$ over the choice of an i.i.d. sample S of size m, for **every** ERM hypothesis h_S, the following holds:

$$L_{(D,f)}(h_S) \leq \epsilon$$

For a sufficiently large m, the ERM_H rule over a finite hypothessi class will be *probably* $(1 - \delta)$ *approximately* (up to error ϵ) correct (PAC).

Chapter 3: A formal learning model

PAC Learning

DEFINITION 3.1 (PAC Learnability) A hypothesis class $\mathcal{H}$ is PAC learnable if tere exists a function $m_{\mathcal{H}} : (0,1)^2 \to \mathbb{N}$ and a learning algorithm with the following property: For every $\epsilon, \delta \in (0,1)$, for every distribution $\mathcal{D}$ over $\mathcal{X}$ and

for every labeling function $f : \mathcal{X} \to \{0, 1\}$, if the realizable assumption holds w.r.t $\mathcal{H}, \mathcal{D}, f$, then when rnning the learning algorithm on $m \geq m_{\mathcal{H}}(\epsilon, \delta)$ i.i.d. examples generated by $\mathcal{D}$ and labeled by f, the algorithm returns a hypothesis h s.t. with probability $1 - \delta$ over the choice of samples, $L_{(\mathcal{D},f)}(h) \leq \epsilon$.

Approximations in PAC Learnability: ϵ defines the distance we allow from the optimal classifier, while δ indicates how likely the classifier is to meet that accuracy requirement (based on a sample from $\mathcal{D}$).

Sample complexity is the *minimal** number of examples formulated as a function of accuracy and confidence required to guarantee a PAC solution. $m_H : (0, 1)^2 \to \mathbb{N}$ or, $m_H : (\epsilon, \delta) \to \mathbb{N}$.

COROLLARY 3.2 Every finite hypothesis class is PAC learnable with sample complexity

$$m_H(\epsilon, \delta) \leq \left\lceil \frac{log(|\mathcal{H}|/\delta)}{\epsilon} \right\rceil$$

Relaxations: Two relaxations with respect to our original problem are necessary for real-world problems:

- Removing the realizability assumption (*agnostic* PAC)
- Learning problems beyond binary classification

Agnostic PAC Learning: practically, it is not realistic that there will exist a classifier which perfectly approximates the labeling function over the whole data distribution. In place of the absolute labelling function f, we introduce the conditional probability $D((x, y)\|x)$ indicating the probability of an input sample x to have the class label y.

Essentially, the constraint that there exists a $h^* \in \mathcal{H}$ s.t. $\mathcal{P}_{x\sim D}[h^*(x) = f(x)] = 1$ *(realizability assumption)* is waived.

Revising the empirical and true error

$$L_D(h) \stackrel{def}{=} \mathop{\mathbb{P}}_{(x,y)\sim\mathcal{D}}[h(x) \neq y] \stackrel{def}{=} \mathcal{D}(\{(x, y) : h(x) \neq y\})$$

Essentially, $f(x)$ is replaced by y, signifying that we now randomly draw *labeled points*.

The definition of empirical risk remains the same as before. Note: $L_S(h) = L_{D\text{uniform over}S}(h)$

The Bayes optimal predictor:

Given any probability distribution $\mathcal{D}$ over $\mathcal{X} \times \{0, 1\}$, the best label predicting function will be:

$$f_{\mathcal{D}}(x) = \begin{cases} 1, & \text{if } \mathcal{P}[y=1|x] \geq 1/2 \\ 0, & \text{otherwise} \end{cases}$$

Read: assign class label 1 to samples that have probability of having label 1 larger than 0.5, and 0 otherwise. However, we do not know $\mathcal{D}$ and we cannot use this optimal predictor, however it can serve as a lower bound on the error of our predictor.

DEFINITION 3.3 (Agnostic PAC Learnability) A *hypothesis class* $\mathcal{H}$ is agnostic PAC learnable if there exists a function $m_{\mathcal{H}} : (0,1)^2 \to \mathcal{N}$ and a learning algorithm with the following property: For **every** $\epsilon, \delta \in (0,1)$ and for **every** distribution $\mathcal{D}$ over $X \times Y$, when running the learning algorithm on $m \geq m_{\mathcal{H}}(\epsilon, \delta)$ i.i.d. examples generated by $\mathcal{D}$, the algorithm returns a hypothesis h s.t. with probability of at least $1 - \delta$ (over the choice of m training examples),

$$L_{\mathcal{D}}(h) \leq \min_{h^* \in \mathcal{H}} L_{\mathcal{D}}(h^*) + \epsilon$$

Agnostic PAC defines the *relative* distance from the best classifier in the chosen hypothesis class rather then the absolute minimal error.

Problems beyond binary classification

- Multiclass classification: the *label set* is no longer binary, but a finite size set (ex. text classification)
- Regression: the *label set* is renamed to the *target set* , and can obtain any value from the set of real numbers.

To accomodate these extensions which mainly only reformulate the loss functio, we introduce *generalized loss functions*.

Generalized loss functions

Given any set $\mathcal{H}$ and a domain set $\mathcal{Z}$, we define the *loss function* l as any function mapping from HxZ to nonnegative real numbers: $l : \mathcal{H} \times \mathbb{Z} \to \mathbb{R}_+$.

Further, we define a *risk function* as the expected loss of a classifier $h \in \mathcal{H}$ w.r.t a probability distribution $\mathcal{D}$ over $\mathbb{Z}$:

$$L_{\mathcal{D}}(h) \stackrel{def}{=} \mathop{\mathbb{E}}_{z \sim \mathcal{D}}[l(h,z)]$$

Similarly, the *empirical risk* is defined as the expected loss over a concrete sample $S = (z_1, \ldots, z_m) \in Z^m$:

$$L_S(h) \stackrel{def}{=} \frac{1}{m} \sum_{i=1}^{m} l(h, z_i)$$

Agnostic PAC learnability for general loss functions

DEFINITION 3.4 (Agnostic PAC for GLF) Everything remains the same as in **3.3** except the loss definition is redefined as: $L_{\mathcal{D}}(h) = \underset{z\sim\mathcal{D}}{\mathbb{E}}[l(h,z)]$