Introduction to Machine Learning **Spring Semester**

Homework 2: June 12, 2024

Due: July 4, 2024

Theory Questions

1. **(15 points) PAC learnability of ℓ_2-balls around the origin.** Given a real number $R \geq 0$ define the hypothesis $h_R : \mathbb{R}^d \to \{0,1\}$ by,

$$h_R(\mathbf{x}) = \begin{cases} 1 & \|\mathbf{x}\|_2 \leq R \\ 0 & \text{otherwise.} \end{cases}$$

Consider the hypothesis class $\mathcal{H}_{ball} = \{h_R \mid R \geq 0\}$. Prove directly (without using the Fundamental Theorem of PAC Learning) that $\mathcal{H}_{ball}$ is PAC learnable in the realizable case (assume for simplicity that the marginal distribution of X is continuous). How does the sample complexity depend on the dimension d? Explain.

2. **(15 points) PAC in expectation**. Consider learning in the realizable case. We say a hypothesis class $\mathcal{H}$ is ***PAC learnable in expectation*** using algorithm A if there exists a function $N(a) : (0,1) \to \mathbb{N}$ such that $\forall a \in (0,1)$ and for any distribution P (realizable by $\mathcal{H}$), given a sample set S such that $|S| \geq N(a)$, it holds that,

$$\mathbb{E}\left[e_P(A(S))\right] \leq a.$$

Show that $\mathcal{H}$ is PAC learnable *if and only if* $\mathcal{H}$ is PAC learnable in expectation (Hint: For one direction, use the law of total expectation. For the other direction, use Markov's inequality).

3. **(10 points) Union of intervals.** Determine the VC-dimension of $\mathcal{H}_k$ - the subsets of the real line formed by the union of k intervals (see the programming assignment for a formal definition of $\mathcal{H}$). Prove your answer.

4. **(10 points) Inhomogeneous linear classifiers.** Prove that the VC-dimension of $\mathcal{H}_d$, the class of inhomogeneous linear classifiers in $\mathbb{R}^d$, is $d+1$. $\mathcal{H}_d$ is the class of all hypotheses of the form

$$h_{\boldsymbol{w},b}(\mathbf{x}) = sign(\boldsymbol{w} \cdot \mathbf{x} - b),$$

where $\boldsymbol{w} \in \mathbb{R}^d$ and $b \in \mathbb{R}$ (Hint: Proceed along the lines of the proof for homogeneous linear classifiers from recitation 3. For the upper bound, given a sample $\mathbf{x}_1, \ldots, \mathbf{x}_{d+2} \in \mathbb{R}^d$, construct a new set of points $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{d+2}$ by appending a constant entry of 1 to each of the $\mathbf{x}_i$'s. What can you say about the new set of points as a subset of $\mathbb{R}^{d+1}$?)

5. **(10 points) Prediction by polynomials.** Given a polynomial $p : \mathbb{R} \to \mathbb{R}$ define the hypothesis $h_p : \mathbb{R}^2 \to \{0, 1\}$ by,

$$h_p(x_1, x_2) = \begin{cases} 1 & p(x_1) \geq x_2 \\ 0 & \text{otherwise.} \end{cases}$$

Determine the VC-dimension of $\mathcal{H}_{\text{poly}} = \{h_p \mid P \text{ is a polynomial}\}$. You can use the fact that given n distinct values $x_1, ..., x_n \in \mathbb{R}$ and $z_1, ..., z_n \in \mathbb{R}$ there exists a polynomial P of degree $n - 1$ such that $P(x_i) = z_i$ for every $1 \leq i \leq n$.

Programming Assignment

1. **Union Of Intervals.** In this question, we will study the hypothesis class of a finite union of disjoint intervals, and the properties of the ERM algorithm for this class.

   To review, let the sample space be $\mathcal{X} = [0,1]$ and consider a binary classification problem, i.e., $\mathcal{Y} = \{0,1\}$. We will try to learn using an hypothesis class that consists of k intervals. More explicitly, let $I = \{[l_1, u_1], \ldots, [l_k, u_k]\}$ be k disjoint intervals, such that $0 \leq l_1 \leq u_1 \leq l_2 \leq u_2 \leq \ldots \leq u_k \leq 1$. For each such k disjoint intervals, define the corresponding hypothesis as

   $$h_I(x) = \begin{cases} 1 & \text{if } x \in [l_1, u_1] \cup \ldots \cup [l_k, u_k] \\ 0 & \text{otherwise} \end{cases}$$

   Finally, define $\mathcal{H}_k$ as the hypothesis class that consists of all hypotheses that correspond to k disjoint intervals:

   $$\mathcal{H}_k = \{h_I | I = \{[l_1, u_1], \ldots, [l_k, u_k]\},\ 0 \leq l_1 \leq u_1 \leq l_2 \leq u_2 \leq \ldots \leq u_k \leq 1\}$$

   We are given a sample of size n: $(x_1, y_1), \ldots, (x_n, y_n)$. Assume that the points are sorted, so that $0 \leq x_1 < x_2 < \ldots < x_n \leq 1$.

   Submission Guidelines:

   - Download the files `skeleton.py` and `intervals.py` from Moodle. You should implement only the missing code in `skeleton.py`, as specified in the following questions. In every method description, you will find specific details on its input and return values.
   - Your code should be written with Python 3.
   - Your submission should include exactly two files: `assignment2.py` (replacing `skeleton.py`) and `intervals.py`.

   Explanation on `intervals.py`:

   The file `intervals.py` includes a function that implements an ERM algorithm for $\mathcal{H}_k$. Given a sorted list $\text{xs} = [x_1, \ldots, x_n]$, the respective labeling $\text{ys} = [y_1, \ldots, y_n]$ and k, the given function `find_best_interval` returns a list of up to k intervals and their error count on the given sample. These intervals have the smallest empirical error count possible from all choices of k intervals or less.

   Note that in sections (c)-(e) you will need to use this function for large values of n. Execution in these cases could take time (more than 10 minutes for an experiment), so plan ahead.

   (a) **(8 points)** Assume that the true distribution $P[x,y] = P[y|x] \cdot P[x]$ is as follows: x is distributed uniformly on the interval $[0,1]$, and

   $$P[y=1|x] = \begin{cases} 0.8 & \text{if } x \in [0, 0.2] \cup [0.4, 0.6] \cup [0.8, 1] \\ 0.1 & \text{if } x \in (0.2, 0.4) \cup (0.6, 0.8) \end{cases}$$

and $P[y=0|x] = 1 - P[y=1|x]$. Since we know the true distribution P, we can calculate $e_P(h)$ precisely for any hypothesis $h \in \mathcal{H}_k$. What is the hypothesis in $\mathcal{H}_{10}$ with the smallest error (i.e., $\arg\min_{h \in \mathcal{H}_{10}} e_P(h)$)?

(b) **(8 points)** Write a function that, given a list of intervals I, calculates the true error $e_P(h_I)$. Then, for $k = 3$, $n = 10, 15, 20, \ldots, 100$, perform the following experiment $T = 100$ times: (i) Draw a sample of size n and run the ERM algorithm on it; (ii) Calculate the empirical error for the returned hypothesis; (iii) Calculate the true error for the returned hypothesis. Plot the empirical and true errors, averaged across the T runs, as a function of n. Discuss the results. Do the empirical and true errors decrease or increase with n? Why?

(c) **(8 points)** Draw a sample of size $n = 1500$. Find an ERM hypothesis for $k = 1, 2, \ldots, 10$, and plot the empirical and true errors as a function of k. How does the error behave? Define k^* to be the k with the smallest empirical error for ERM. Does this mean the hypothesis with k^* intervals is a good choice?

(d) **(8 points)** Now we will use the principle of structural risk minimization (SRM), to search for a k that gives a good test error. Let[1] $\delta_k = \frac{0.1}{k^2}$.

- Use to following penalty function:

$$2\sqrt{\frac{\text{VCdim}(\mathcal{H}_k) + \ln \frac{2}{\delta_k}}{n}}$$

- Draw a data set of $n = 1500$ samples, run the experiment in (c) again, but now plot two additional lines as a function of k: 1) the penalty, and 2) the sum of penalty and empirical error.
- What is the best value for k in each case? is it better than the one you chose in (c)?

(e) **(8 points)** Here we will use holdout-validation to search for a $k \in \{1, ..., 10\}$ that gives good test error. Draw a data set of $n = 1500$ samples and use 20% for a holdout-validation. Choose the best hypothesis and discuss how close this gets you to finding the hypothesis with optimal true error. No need to retrain on the entire dataset; you can simply use the hypothesis obtained during the holdout-validation process.

[1] See the notes of lecture #3 for more details regarding a similar setting of δ_k.