Question 1.

Consider the Winnow algorithm.

(a) What concept class was Winnow designed for? What is Winnow's mistake bound for that class?

(b) Adapt the algorithm to learn general conjunctions. How will the mistake bound change?

Question 2.

Consider the halving algorithm with hypothesis class (initial hypothesis) $\mathcal{H}_1$ of all non-contradictory conjunctions on 3 propositional variables.

(a) Determine $|\mathcal{H}_1|$.

(b) Give an upper bound on $|\mathcal{H}_2|$ given that first prediction was incorrect.

Question 3.

Consider halving algorithm with the initial version space $\mathcal{H}$ consisting

(a) of all conjunctions of exactly 3 different non-negative literals, i.e.,

$$\mathcal{H} = \{\, p_i \wedge p_j \wedge p_k \mid 1 \leq i < j < k \leq n \,\}$$

(b) of all conjunctions that use some of the given variables (and the empty conjunction).

(c) of all n-CNFs.

1. For each scenario, determine if the learner learns $\mathcal{H}$ online (in the mistake-bound model) and justify your answer.
2. For each scenario where the learner learns in the mistake bound model, decide if they learn efficiently as well. Assume that checking the consistency of a single hypothesis with an observation takes a unit of time.
3. For the **first** case, assume the first example is $(0, 1, 1, 1, \dots 1)$ and it is a negative instance. What will be the learner's prediction for the second example, which is $(0, 1, 0, 1, \dots)$? Justify your answer.

Question 4.

Consider the following hypothesis classes. For each hypothesis class, determine its VC dimension and provide a brief proof.

(a) $\mathcal{H} = \{h : \mathbb{R} \mapsto \{0; 1\}, h(x) = [\![x > t]\!], t \in \mathbb{R}\}$

(b) $\mathcal{H} = \{h : \mathbb{R} \mapsto \{0; 1\}, h(x) = [\![t_1 \leq x < t_2]\!], t_1 < t_2 \in \mathbb{R}\}$

(c) $\mathcal{H}$ is the set of all monotone conjunctions on n variables.