10-1

Let $\varepsilon, \delta \in (0,1)$. Pick k "chunks" of size $m_H(\varepsilon/2)$.

Apply A on each of these chunks, to obtain $\hat{h}_1, \hat{h}_2, \ldots, \hat{h}_k$.

Note that the probability that $\min_{i \in [k]} L_D(\hat{h}_i) \leq \min L_D(h) + \varepsilon/2$

is at least $1 - \delta_0^k \geq 1 - \delta/2$.

Now apply an ERM over the class $\hat{H} := \{\hat{h}_1, \ldots, \hat{h}_k\}$

with the training data being the last chunk of size $\left\lceil \frac{2\log(\frac{4k}{\delta})}{\varepsilon^2} \right\rceil$

Denote the output hypothesis by $\hat{h}$. Using Corollary 4.6

we obtain that with probability at least $1 - \delta/2$

$$L_D(\hat{h}) \leq \min_{i \in [k]} L_D(\hat{h}_i) + \frac{\varepsilon}{2}$$

Applying the Union bound we obtain that with probability at least $1-\delta$,

$$L_D(\hat{h}) \leq \min L_D(\hat{h}_i) + \varepsilon/2 \leq \min_{h \in H} L_D(h) + \varepsilon$$

10-4

a) Let X be a finite set of size n. Let B be the class of all functions from X to $\{0,1\}$. Then, $L(B,T)=B$ and both are finite. Hence for any $T\geq 1$

$$\mathrm{VCdim}(B)=\mathrm{VCdim}(L(B,T))=\log 2^n=n$$

b) Denote by β the class of decision stumps in $\mathbb{R}^d$. Formally,

$$\beta=\{h_{j,b,\theta} : j\in[d], b\in\{-1,1\}, \theta\in\mathbb{R}\}, \text{ where } h_{j,b,\theta}(X)=b\cdot\mathrm{sign}(\theta-x_j)$$

Note that $\mathrm{VCdim}(\beta_j)=2$

Clearly, $\beta=\bigcup_{j=1}^{d}\beta_j$. Applying Exercise 11, we conclude that

$$\mathrm{VCdim}(\beta)\leq 16+2\log d$$

Assume w.l.o.g that $d=2^k$ for some $k\in\mathbb{N}$ (o.w $d=2^{\lfloor\log d\rfloor}$).
Let $A\in\mathbb{R}^{k\times d}$ be the matrix whose columns range over the entire set $\{0,1\}^k$. For each $i\in[k]$, let $x_i=A_{i\rightarrow}$.
we claim that the set $C=\{x_i,\dots,x_k\}$ is shattered.
Let $I\subseteq[k]$. We show that we can lable the instances in I positively, while the instances $[k]\setminus I$ are lableed negatively. By over construction, ther exisits an index j such that $A_{ij}=X_{i,j}=1$ iff $i\in I$. Then $h_{j,-1,\frac{1}{2}}(x_i)=1$ iff $i\in I$

c) Following the hint, for each $i \in [Tk/2]$, let $x_i = \lceil i/k \rceil A_{i \rightarrow}$. We claim that set $C = \{x_i : i \in [Tk/2]\}$ is shattered by $L(B_d, T)$.

Let $I \subseteq [Tk/2]$. Then $I = I_1 \cup I_2 \cdots \cup I_{T/2}$, where each I_t is a subset of $\{(t-1)k+1, \ldots, tk\}$. For each $t \in [T/2]$, let j_t be the corresponding column of A (i.e., $A_{i,j} = 1$ iff $(t-1)k+i \in I_t$)

Let

$$h(x) = \text{Sign}\Big(h_{j_1,-1,1/2} + h_{j_1,1,3/2} + h_{j_2,-1,3/2} + h_{j_2,1,5/2} + \cdots$$

$$+ h_{j_{T/2-1},-1,T/2-3/2} + h_{j_{T/2-1},1,T/2-1/2} + h_{j_{T/2},-1,T/2-1/2}\Big)(x)\Big)$$

Then $h(x_i) = 1$ iff $i \in I$, finally observe that $h \in L(B_d, T)$.

11.1

Let S be an i.i.d. Sample. Let h be the output of the described learning algorithm. Note that (independently of the identity of S),

$L_D(h) = 1/2$ (since h is a constant function).

Let us calculate the estimate $L_V(h)$. Assume that the parity of S is

1. Fix some fold $\{(x,y)\} \subseteq S$. We distinguish between two ~~cases~~ cases:

   - The parity of $S \setminus \{x\}$ is 1. It follows that $y = 0$. When being trained using $S \setminus \{x\}$, the algorithm outputs the constant predictor $h(x) = 1$. Hence, the leave-one-out estimate using this fold is 1.

   - The parity of $S \setminus \{x\}$ is 0. It follows that $y = 1$ when being trained using $S \setminus \{x\}$, the algorithm outputs the constant predictor $h(x) = 0$. Hence, the leave-one-out estimate using this fold is 1

Averaging over the folds, the estimate of the error of h is 1. Consequently, the difference between the estimate and true error is $1/2$. The case in which the parity of S is analyzed analogously.

11.2 Consider for example the case in which $H_1 \subseteq H_2 \subseteq \cdots \subseteq H_K$ and $|H_i| = 2^i$ for every $i \in K$. Learning H_K in the Agnostic-Pac model provides the following bound for an ERM hypothesis h:

$$L_D(h) \leq \min_{h \in H_K} L_D(h) + \sqrt{\frac{2(k+1+\log 1/\delta)}{m}}$$

Alternatively, we can use model selection as we describe next. Assume that j is the minimal index which contains a hypothesis $h^* \in \operatorname{argmin}_{h \in H} L_D(h)$. Fix some $r \in [K]$. By Hoffding's inequality, with probability at least $1-\delta/(2K)$ we have

$$|L_D(\hat{h}_r) - L_V(\hat{h}_r)| \leq \sqrt{\frac{1}{2\alpha m} \log 4/\delta}$$

Applying the union bound, we obtain that with probability at least $1-\delta/2$, the following inequality holds (Simultaneously) for every $r \in [K]$:

$$L_D(\hat{h}) \leq L_V(\hat{h}) + \sqrt{\frac{1}{2\alpha m} \log \frac{4K}{\delta}} \leq L_V(\hat{h}) + \sqrt{\frac{1}{2\alpha m} \log \frac{4K}{\delta}}$$

$$\leq L_D(\hat{h}_r) + 2\sqrt{\frac{1}{2\alpha m} \log \frac{4K}{\delta}} = L_D(\hat{h}_r) + \sqrt{\frac{2}{\alpha m} \log \frac{4K}{\delta}}$$

In particular with probability at least $1-\delta/2$

$$L_D(\hat{h}_j) \leq L_D(h^*) + \sqrt{\frac{2}{(1-\alpha)m} \log \frac{4|H_j|}{\delta}} = L_D(h^*) + \sqrt{\frac{2}{(1-\alpha)m} \log \frac{4|H_j|}{\delta}}$$

Combining the two last inequalities with the union bound we obtain that with probability at least $1-\delta$

$$L_D(\hat{h}) \leq L_D(h^*) + \sqrt{\frac{2}{\alpha m}\log\frac{4K}{\delta}} + \sqrt{\frac{2}{(1-\alpha)m}\log\frac{4|H_j|}{\delta}}$$

We conclude that

$$L_D(\hat{h}) \leq L_D(h^*) + \sqrt{\frac{2}{\alpha m}\log\frac{4K}{\delta}} + \sqrt{\frac{2}{(1-\alpha)m}\left(j+\log\frac{4}{\delta}\right)}$$

Comparing the two bounds, we see that when the "optimal index" j is significantly smaller than K, the bound achieved using model selection is much better. Being even more concrete, if j is logarithmic in K, we achieve a logarithmic improvement.

18.2

We denote by H the binary entropy:

(a) The algorithm first picks the root node, by searching for the feature which maximizes the information gain.

The information gain for feature 1 (namely, if choose $x_1=0$? as the root) is:

$$H(1/2) - \left(\tfrac{3}{4} H(2/3) + \tfrac{1}{2} H(0)\right) \simeq 0.22$$

The information gain for feature 2, as well as feature 3 is: $H(1/2) - \left(\tfrac{1}{2} H(1/2) + \tfrac{1}{2} H(1/2)\right) = 0$

So the algorithm picks $x_1=0$? as the root. But this means that the three examples $((1,1,0),0)$ $((1,1,1),1)$ and $((1,0,0),1)$ go down one subtree, and no matter what question we'll ask now, we won't be able to classify all three examples perfectly. For instance, if the next question is $x_2=0$? (after which we must give a prediction), either $((1,1,0),0)$ or $((1,1,1),1)$ will be mislabeled. So in any case at least one example will be mislabeled. Since we have four examples in the training set, it follows that the training error is at least $1/4$

b) Here is one such tree: