3.2

X: دامنه است

$$H = \{h_z : z \in X\} \cup \{h^-\}$$

$$\text{for} \quad z \in X \qquad h_z(x) = \begin{cases} 1 & x = z \\ 0 & x \neq z \end{cases}$$

h^- : همه نمونه‌ها را منفی برچسب می‌زند $\quad \forall x \in X \quad h^-(x) = 0$

فرض realizability : این فرض می‌گوید تابع درست f به همه نمونه‌ها برچسب درستی می‌زند به جز یکی.

الف) الگوریتم ERM ارائه دهید.

با توجه به فرض realizability طبق این سوال تابع f تابعی است که به همه xها به جز یکی برچسب منفی (0) می‌دهد اگر اسم این x را بگذاریم x^* یعنی همه xها به صورت $(x,0)$ هستند اما x^* به صورت $(x^*,1)$ است.

طبق تعریف $h_z(x)$ فقط زمانی که $x=z$ باشد برچسب 1 می‌دهد پس اگر در مجموعه داده‌ها آموزشی x^* بیاید و ما بتوانیم h_{x^*} اجرا شود یعنی فقط برای x^* برچسب 1 می‌دهد و برای بقیه برچسب صفر می‌دهد، و این یعنی برچسب همه xها و x^* درست گذاشته شوند.

و اگر زمانی که x^* در مجموعه آموزشی نیاید و الگوریتم h^- را برگرداند یعنی برچسب همه xها صفر گذاشته شود و این هم اینبار درست است چون همه xها برچسب صفر می‌گیرند

پس الگوریتم A را می‌توانیم اینگونه معرفی کنیم که اگر x^* در داده بیاید $h_z(x^*)$ را برگرداند و اگر x^* در داده نیاید h^- را برگرداند.

همانطور که دیدیم طبق این الگوریتم در هر دو صورت برچسب گذاری‌ها درست انجام می‌شود پس این الگوریتم یک الگوریتم ERM است یعنی کمترین خطای تجربی را دارد (خطا صفر است).

ادامه 3.2

پس باید نشان دهیم H، PAC learnable است.

پس باید نشان دهیم با احتمال $1-\delta$ خطای حقیقی کمتر از ε است $\varepsilon, \delta \in (0,1)$

به عبارتی اگر $D^m(\{S|_m \quad L_{D,f}(h) > \varepsilon\}) < \delta$ باشد خوب است.

در این سوال برای محدود نمودن خطا باید روی x^* تمرکز کنیم.

اگر احتمال حضور x^* در D کمتر از ε باشد و x^* در S نیاید خطای حقیقی همچنان کمتر از ε است ولی اگر احتمال حضور x^* در D بیشتر از ε باشد و x^* در S نیاید مشکل ساز شده و خطا زیاد است یعنی همه نمونه‌های منفی هستند و ماشین x^* را ندارد درحالی که احتمال وجود x^* در D کما بیشتر از ε است.

$D(x^* \in S) > \varepsilon$

$$\Rightarrow D^m(\{S|_m \quad L_{D,f}(h_S) > \varepsilon\}) = D^m(x^* \notin S) \leqslant (1-\varepsilon)^m \leqslant e^{-\varepsilon m}$$

$$\Rightarrow m_H(\varepsilon, \delta) \leqslant \left\lceil \frac{\log \frac{1}{\delta}}{\varepsilon} \right\rceil$$

3.3 $X=\mathbb{R}^2$ $\quad Y=\{0,1\}$ $\quad$ h: کلاس دایره‌ها متحدالمرکز در صفر

$$H=\{h_r : r\in\mathbb{R}^+\} \quad \text{where} \quad h_r(x)=1_{\{\|x\|\le r\}}$$

$$h_r(x)=\begin{cases}1 & \|x\|\le r\\ 0 & o.w\end{cases}$$

این مسئله در صورت realizability می‌خواهیم ثابت کنیم مجموعه فرضیات H، PAC learnable است و

$$m_H(\epsilon,\delta)\overset{?}{\le}\frac{\log(1/\delta)}{\epsilon}$$

الگوریتم ERM که A را در نظر می‌گیریم که به ازای مجموعه داده‌های آموزشی $S=(x_i, y_i)\ \ i=1,\dots,m$ ، $h(S)$ ای را برمی‌گرداند که مربوط به کوچکترین دایره‌ای است یعنی این دایره همهٔ نمونه‌های مثبت را در بر دارد. فرضیهٔ realizability برقرار است یعنی وجود دارد یک $h_{r^*}\in H$ که خطای حقیقی آن روی توزیع X صفر باشد.

$$L_{D,f}(h_{r^*})=0$$

پس همواره $r\le r^*$ است.

برای نشان دادن PAC le- باید نشان دهیم برای تابع m_H ، $P(\{S|_x : L_{D,f}\,h(S)>\epsilon\})<\delta$

یعنی خطای حقیقی با احتمال کمتر از δ بیشتر از ϵ است. و این خوب است.

پس زمانی خطا داریم که $L_{D,f}\,h(S)>\epsilon$ باشد.

اگر برای یک ϵ در نظر بگیریم r' را طوری که $P(\{x : \overbrace{r'\le\|x\|<r^*}^{\sim}\})=\epsilon$

یعنی احتمال اینکه $\|x\|$ بین r' و r^* باشد را ϵ بگیریم.

ادامه سؤال 3.3

ما می‌خواهیم احتمال جاهایی را پیدا کنیم که $L_{D,f}h(s) > \varepsilon$ است پس یعنی جاهایی که h که در N نباشد ($S \cap N = \phi$)

$$\Rightarrow P^m(\{S|_x : L_{D,f}h(s) > \varepsilon\}) = P^m(\{S|_x : S \cap N = \phi\})$$

$$\overset{\text{استقلال بین }x\text{ها}}{=} \prod_{i=1}^{m} P(\{S|_x : S \setminus N\}) \overset{①}{=} \prod_{i=1}^{m} (1-\varepsilon) = (1-\varepsilon)^m$$

① چون احتمال برای x اتفاق این که x در که باشد ما در N نباشد برابر $1-\varepsilon$ است.

طبق بسط تیلور: $1+a \simeq e^a \Leftarrow (1-\varepsilon)^m \leqslant e^{-\varepsilon m}$

$$\Rightarrow P(\{S|_x : L_{D,f}h(s) > \varepsilon\}) \leqslant e^{-\varepsilon m}$$

$$\Rightarrow P(\{S|_x : L_{D,f}h(s) > \varepsilon\}) < \delta \equiv e^{-\varepsilon m} \leqslant \delta$$

$$\equiv m_H(\varepsilon, \delta) \leqslant \left\lceil \frac{\log(1/\delta)}{\varepsilon} \right\rceil$$

سؤال 3.4

با توجه به صورت مسئله H متناهی است و می‌دانیم هر مجموعه فرضیه متناهی PAC learnable است

با $m \geq \frac{\log \frac{|H|}{\delta}}{\epsilon}$ برای $\epsilon , \delta \in (0,1)$

$|H|$ یعنی اندازه H برای d می‌شود $3^d + 1$ چون ۳ حالت برای هر i وجود دارد یا x_i باید باشد یا $\bar{x}_i = 1 - x_i$ باید باشد یا هیچ کدام نباید و به علاوه یک حالت فرضیه تماماً منفی که به صورت سؤال گفته شده

پس

چون لگاریتم تابعی صعودی است

$$m \geq \frac{\log \frac{3^d + 1}{\delta}}{\epsilon} \Rightarrow m \geq \frac{\log \left(\frac{3^d}{\delta} + \frac{1}{\delta} \right)}{\epsilon} \geq \frac{\log \frac{3^d}{\delta}}{\epsilon}$$

$$\Rightarrow m \geq \frac{d \log \frac{3}{\delta}}{\epsilon} \Rightarrow m \geq \frac{d \log 3 + \log \frac{1}{\delta}}{\epsilon}$$

$$m \geq \frac{d \log 3 - \log \delta}{\epsilon}$$

به عبارتی

$$m_H(\epsilon, \delta) \leq \left\lceil \frac{d \log 3 - \log \delta}{\epsilon} \right\rceil$$

3.5

X: دامنه ، $D_1, \dots, D_n$ دنباله‌ای از توزیع‌ها روی X

H مجموعه‌ای متناهی از دنباله‌ای از توابع دودویی روی X ، $f \in H$

فرض می‌کنیم یک نمونه m تایی S را که دریافت می‌کنیم به طوری که x_i ها مستقل هستند اما هم‌توزیع نیستند.

این صورت (x_i) نمونه‌ای است از توزیع D_i ، $y_i = f(y_i)$ ، $\bar{D}_m = \frac{\sum D_i}{m}$

ثابت می‌کنیم برای یک $\epsilon \in (0,1)$ را ثابت می‌کنیم

$$P[\exists h \in H \ \text{ s.t } \ L_{\bar{D},f}(h) > \epsilon \ \text{ and } \ L_{S,f}(h) = 0] \leq |H| e^{-\epsilon m}$$

$$N = \{S|_x \ L_{\bar{D},f}(h) > \epsilon\} \qquad H_B = \{h \in H : L_{\bar{D},f}(h) > \epsilon\}$$

$$M = \{S|_x \ L_S(h) = 0\} \implies N \subseteq M$$

$$\implies P(N) \leq \sum_{h \in H_B} P(\{S|_x \ L_S(h) = 0\})$$

$$P(\{S|_x \ L_S(h) = 0\}) = P(\{S|_x \ \forall_{x \sim D_i} \ h(x) = f(x)\})$$

(استقلال)

$$\underline{\underline{\text{i.e.}}} \ \prod_{i=1}^{m} P(\{S|_{x \sim D_i} \ h(x) = f(x)\}) = \prod_{i=1}^{m} \left(P(\{S|_{x \sim D_i} \ h(x) = f(x)\})^{\frac{1}{m}}\right)^m$$

$$\overset{*}{\leq} \prod_{i=1}^{m} \left(\frac{\sum P_{x \sim D_i}(\{S|_x \ h(x) = f(x)\})}{m} \right)^m \overset{**}{\leq} \prod_{i=1}^{m} (1-\epsilon) = (1-\epsilon)^m \leq e^{-\epsilon m}$$

$$\implies P(N) \leq \sum e^{-\epsilon m} = |H_B| e^{-\epsilon m} \leq |H| e^{-\epsilon m} \implies$$

چون برای حالت عام همه $h \in H$ است (H_B در تمام H است.)

فصل سوم — بازگیری ماشین — فاطمه داوری

ادامه 3.5

$$\Rightarrow P(\{S|_x : L_{(D,f)}(h) \leq \varepsilon\}) \leqslant |H| e^{-\varepsilon m}$$

$$G = \sqrt[m]{\prod a_i} \leqslant \bar{a} = \frac{\sum a_i}{m}$$

* ← دلیل این نامساوی رابطه بین میانگین حسابی و هندسی است

** دلیل این جایگذاری این است که

$$\frac{\sum_{x_i \sim D_i} P(\{S|_x h(x_i) = f(x_i)\})}{m} = \frac{P_{x \sim D_1}(\{h(x) = f(x)\}) + \dots + P_{x \sim D_m}(\{h(x) = f(x)\})}{m}$$

$$\leqslant 1 - \varepsilon$$

چون تعریف کردیم $L_{\bar{D},f}(h) > \varepsilon$

پس می‌توان گفت به ازای هر $\varepsilon \in (0,1)$ داریم

$$P[\exists h \in H \quad s.t \quad L_{\bar{D},f}(h) > \varepsilon \quad \text{and} \quad L_{S,f}(h) = 0] \leqslant |H| e^{-\varepsilon m}$$

✓

3.6

همانطور که ثابت می‌کنیم یک مجموعه فرضیات H از طبقه‌بندی‌های دودویی اگر Agnostic PAC-learnable باشد، PAC le... است.

اگر A یک الگوریتم موفق برای Agnostic... است برای PAC le... هم یک الگوریتم موفق است.

طبق تعریف گوییم H، Agnostic PAC... است اگر وجود داشته باشد یک تابع $m_H(\epsilon,\delta) \to \mathbb{N}$ گوییم A وجود دارد با این خاصیت برای هر $(\delta,\epsilon) \in (0,1)$ و هر توزیع X، Y

خروجی الگوریتم ما یک $h(S)$ برهد که با احتمال $1-\delta$ اضافی موفق این تابع از مقدار $\min_{h' \in H} L_D(h') + \epsilon$ کمتر باشد یعنی

$$L_D(h) \leq \min_{h' \in H} L_D(h') + \epsilon$$

$\min_{h' \in H} L_D(h')$ یعنی می‌دانیم یک $h' \in H$ وجود دارد که کمترین خطای واقعی را دارد ولی ما آن را نمی‌دانیم.

اما اگر فرض realizability را در نظر بگیریم یعنی می‌دانیم یک $h^* \in H$ وجود دارد که $L_D(h^*) = 0$ است پس

$h^* = h'$ است و $\min_{h' \in H} L_D(h') = 0$ پس داریم

$$L_D(h) \leq 0 + \epsilon$$

یعنی H، PAC learnable است و الگوریتم A برای PAC learning هم جواب می‌دهد.

3.7

در این سوال باید نشان دهیم پیش بینی کننده بهینه بیز f_D، بهترین استراتژی برای هر طبقه بندی کننده است

روی $X \to [0,1]$ داریم:

$$L_D(f_D) \le L_D(g)$$

پیش بینی کننده بیز به این صورت تعریف می‌شود:

$$f_D(x) = \begin{cases} 1 & \text{if } P[Y=1|x] \ge \frac{1}{2} \\ 0 & o.w \end{cases}$$

$$L_D(f_D) = P(f(x) \neq y | x)$$

$$= P(f(x)=1|x) \cdot P(Y=0|x) + P(f(x)=0|x) \cdot \overbrace{P(Y=1|x)}^{\alpha}$$

$$= P(f(x)=1|x) \cdot (1-\alpha) + P(f(x)=0|x) \cdot \alpha$$

طبق تعریف f میدانیم اگر $\alpha = P[Y=1|x] \ge \frac{1}{2}$ باشد، $f_D(x)$ ۱ است

$$L_D f(x) = \begin{cases} 1\,(1-\alpha) + 0 & \text{if } \alpha \ge \frac{1}{2} \\ 0\,(1-\alpha) + 1\,\alpha & o.w \end{cases} \rightarrow \begin{cases} 1-\alpha & \text{if } \alpha \ge \frac{1}{2} \\ \alpha & \alpha < \frac{1}{2} \end{cases}$$

$$\Rightarrow L_D(f_D) = \min\{1-\alpha, \alpha\}$$

$$L_D(g) = P(g(x)=1|x) \cdot P(Y=0|x) + P(g(x)=0|x) \cdot P(Y=1|x)$$

$$= P(g(x)=1|x) \cdot (1-\alpha) + P(g(x)=0|x) \cdot \alpha$$

$$\ge P(g(x)=1|x) \cdot \min(\alpha, 1-\alpha) + P(g(x)=0|x) \cdot \min(\alpha, 1-\alpha)$$

$$= \min(\alpha, 1-\alpha) \left[P(g(x)=1|x) + P(g(x)=0|x) \right] = \min(\alpha, 1-\alpha)$$

ادامه 3.7

بنابراین

$$L_D(f_D) = \min(\alpha, 1-\alpha)$$

$$L_D(g) \geq \min(\alpha, 1-\alpha) \Rightarrow L_D(f_D) \leq L_D(g)$$

سؤال 5-2 فصل پنجم یادگیری ماشین.

۱- مسئله انتخاب متغیر یک مسئله خیلی مهم است به دلایل زیر:
زمانی که همه متغیرها را در نظر بگیریم ممکن است با دور ریختن متغیرهای دیگر اطلاعات مفیدی را که در آن متغیرها موجود داشته از دست بدهیم. از طرفی اگر بخواهیم همه متغیرهای موجود در مدل داشته باشیم وجود متغیری که اطلاعات مفیدی نمی‌دهد در مدل باعث افزایش خطای تخمین (ϵ_{est}) می‌شود.

می‌دانیم اگر H بزرگ باشد یعنی مجموعه همه توابع را در نظر بگیریم و این خوب نیست چون این مشکل پیش می‌آید که دست ما برای انتخاب باز باشد و انتخاب بین آنها سخت باشد و ممکن است اشتباها انتخاب شود. بنابراین در این صورت خطای تخمین زیاد می‌شود (ϵ_{est}) هرچند خطای تقریب ϵ_{app} صفر می‌شود. و ممکن است بیش برازش رخ دهد. ولی اگر یک سوگیری داشته باشیم و از دانش اولیه برای انتخاب H استفاده کنیم از بیش برازش می‌توانیم جلوگیری کنیم یعنی متعادل این نباشیم که الگوریتم روی مجموعه داده‌های آموزشی خوب عمل کند بلکه برای داده‌های آزمایشی که اطلاعات آنها را ندارد الگوریتم مورد نظر هم بتواند خوب عمل کند. پس خوبی کوچک کردن H این است که ϵ_{est} کمتری دارد ولی باید مراقب باشیم ϵ_{app} خیلی زیاد نباشد. به همین علت می‌توانیم چنین مقایسه کنیم:

$$\epsilon_{app\,H_5} < \epsilon_{app\,H_2} \quad \text{و} \quad \epsilon_{est\,H_5} > \epsilon_{est\,H_2}.$$

۲- می‌دانیم هرچه m بیشتر باشد خطای بالاکرد (ϵ_{est}) کمتر است یعنی L_S به L_D نزدیک‌تر است پس هرچه m کوچک‌تر باشد ممکن است ϵ_{est} بزرگ شود برای کاهش ϵ_{est} ها منطور که گفتیم کوچک کردن H با کاهش متغیرها می‌تواند مفید باشد. پس برای m کوچک استفاده از H_2 مفیدتر است.