



تمرین سری چهارم

موعده تحویل: ۵ آذر

دسته بندها

پرسش ۱

از معروف ترین و مهم ترین دسته بندهای خطی، دسته بند Perceptron می باشد که شبه کد آن در زیر آمده است. الگوریتم Perceptron یک الگوریتم تکرار شونده است که توالی از بردارهای w_1, w_2, \dots را می سازد. به عنوان مقداردهی اولیه، w_1 را بردار تمام صفر قرار می دهیم. در مرحله t ، اگر داده i وجود داشته باشد که به طوری که توسط w_t اشتباه دسته بندی شود یا به عبارت دقیقتر $\text{sign}(\langle w_t, x_i \rangle) \neq y_i$ باشد، آنگاه مرحله ی بروزرسانی بردار w_t به صورت $w_{t+1} \leftarrow w_t + y_i x_i$ انجام می شود. قاعدتا هدف نهایی ما این است که به w_T برسیم به گونه ای که $\forall i : y_i \langle w_T, x_i \rangle > 0$ باشد، یعنی هیچ اشتباهی روی داده هایی که در اختیار داریم نداشته باشد. برای سادگی فرض می کنیم داده هایی که در اختیار داریم با دسته بند خطی، قابل جدا شدن هستند. حالت کلی و در نظر نگرفتن این شرط، مساله ی جالب دیگری است که پیشنهاد می کنم روی آن فکر کنید!

Batch Perceptron

```

input: A training set  $(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_m, y_m)$ 
initialize:  $\mathbf{w}^{(1)} = (0, \dots, 0)$ 
for  $t = 1, 2, \dots$ 
  if  $(\exists i \text{ s.t. } y_i \langle \mathbf{w}^{(t)}, \mathbf{x}_i \rangle \leq 0)$  then
     $\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} + y_i \mathbf{x}_i$ 
  else
    output  $\mathbf{w}^{(t)}$ 

```

الف) پارامتر B را به صورت $B = \min\{\|\mathbf{w}\|_2 : \forall i \in [m], y_i \langle \mathbf{w}, \mathbf{x}_i \rangle \geq 1\}$ و همچنین پارامتر R را به صورت $R = \max_i \|\mathbf{x}_i\|_2$ تعریف می کنیم. اثبات کنید که این الگوریتم حداکثر $(RB)^2$ مرتبه اجرا می شود.

ب) اگر مرحله ی بروزرسانی را به صورت $w_{t+1} \leftarrow w_t + \eta y_i x_i$ انجام دهیم که η یک ابرپارامتر است که عددی نامنفی است، اثبات کنید تعداد مراحل که تا همگرایی طی می شود برای هر دو حالت الگوریتم یکسان خواهد بود.

پرسش ۲

الگوریتم Adaboost را در نظر بگیرید. شبکه‌کد این الگوریتم در زیر آمده است.

```

ADABOOST( $S = ((x_1, y_1), \dots, (x_m, y_m))$ )
1  for  $i \leftarrow 1$  to  $m$  do
2       $\mathcal{D}_1(i) \leftarrow \frac{1}{m}$ 
3  for  $t \leftarrow 1$  to  $T$  do
4       $h_t \leftarrow$  base classifier in  $\mathcal{H}$  with small error  $\epsilon_t = \mathbb{P}_{i \sim \mathcal{D}_t} [h_t(x_i) \neq y_i]$ 
5       $\alpha_t \leftarrow \frac{1}{2} \log \frac{1-\epsilon_t}{\epsilon_t}$ 
6       $Z_t \leftarrow 2[\epsilon_t(1-\epsilon_t)]^{\frac{1}{2}}$   $\triangleright$  normalization factor
7      for  $i \leftarrow 1$  to  $m$  do
8           $\mathcal{D}_{t+1}(i) \leftarrow \frac{\mathcal{D}_t(i) \exp(-\alpha_t y_i h_t(x_i))}{Z_t}$ 
9   $f \leftarrow \sum_{t=1}^T \alpha_t h_t$ 
10 return  $f$ 

```

الف) اگر دقت کنید در الگوریتم Adaboost در هر مرحله یک دسته‌بند به صورتی که طبق توزیع آن مرحله کمترین خطا را داشته باشد انتخاب می‌شود. اثبات کنید این الگوریتم هیچگاه دو تابع یکسان در دو مرحله متوالی انتخاب نمی‌کند ($h_t \neq h_{t+1}$).

ب) بردار توزیع ($D_{t+1}(1), D_{t+1}(2), \dots, D_{t+1}(m)$) را در الگوریتم Adaboost در نظر بگیرید که m تعداد داده‌ها است. اثبات کنید این بردار و برداری که مولفه‌های آن $y_i h_t(x_i)$ هستند، uncorrelated هستند (به این معنی که ضرب داخلی آن‌ها صفر است).

پرسش ۳

یک مسأله‌ی دسته‌بندی باینری را در نظر بگیرید. هر دسته‌بند h در چارچوب این مسأله، تابعی از \mathcal{X} به $\{0, 1\}$ یعنی $h: \mathcal{X} \rightarrow \{0, 1\}$ است. نرخ خطا را به صورت زیر تعریف می‌کنیم.

$$R(h) = \mathbf{P}[Y \neq h(X)]$$

و قاعدتا به همین ترتیب نرخ خطای تجربی نیز به صورت زیر تعریف می‌شود.

$$\hat{R}(h) = \frac{1}{n} \sum_i \mathbf{I}(h(X_i) \neq Y_i)$$

الف) اثبات کنید که تابع h ‌ای که $R(h)$ کمینه کند به صورت زیر است.

$$h^*(x) = \begin{cases} 1 & \text{if } m(x) > \frac{1}{2} \\ 0 & \text{otherwise} \end{cases}$$

که تابع m برابر $m(x) = \mathbb{E}(Y | X = x) = \mathbb{P}(Y = 1 | X = x)$ است. حال فرض کنید که \hat{m} تخمینی از m باشد و مطابق با آن \hat{h} را به صورت زیر تعریف کنیم.

$$\hat{h}(x) = \begin{cases} 1 & \text{if } \hat{m}(x) > \frac{1}{2} \\ 0 & \text{otherwise} \end{cases}$$

ب) با فرض اینکه $R^* = R(h^*)$ است، اثبات کنید

$$R(\hat{h}) - R^* \leq 2 \int |\hat{m}(x) - m^*(x)| P_X(x) dx$$

پرسش ۴

می‌خواهیم با استفاده از درخت تصمیم‌گیری، دسته‌بندی طراحی کنیم که برحسب ویژگی‌های هر رستوران پیش‌بینی کند که آیا از رستوران رضایت خواهیم داشت یا نه. داده‌هایی که جمع‌آوری کرده‌ایم را در شکل زیر مشاهده می‌کنید.

رستوران	نوع غذا	رنج قیمت	محل رستوران	محدودیت‌ها	رضایت‌مندی
1	فست فود	ارزان	غرب شهر	Vegeterian	0
2	ایرانی	متوسط	شمال شهر	Gluten free	0
3	آسیایی	متوسط	شمال شهر	هیچی	0
4	آسیایی	گران	شرق شهر	Vegeterian	0
5	آسیایی	ارزان	غرب شهر	Vegeterian	1
6	فست فود	متوسط	شمال شهر	هیچی	1
7	ایرانی	ارزان	شمال شهر	هیچی	1
8	آسیایی	ارزان	شرق شهر	Gluten free	0
9	فست فود	گران	غرب شهر	هیچی	0
10	ایرانی	متوسط	شرق شهر	Vegeterian	1
11	آسیایی	متوسط	شرق شهر	Gluten free	1

الف) با استفاده از داده‌های داده‌شده، درخت تصمیم‌گیری رضایت‌نداشتن یا نداشتن از رستوران را بسازید. سعی کنید کل مراحل روند حل را توضیح دهید.
 ب) حال با استفاده از دسته‌بندی که ساخته‌اید، رضایت‌مندی پنج رستوران زیر را پیش‌بینی کنید.

رستوران	نوع غذا	رنج قیمت	محل رستوران	محدودیت‌ها
12	فست فود	ارزان	شمال شهر	هیچی
13	ایرانی	متوسط	شرق شهر	هیچی
14	ایرانی	ارزان	غرب شهر	Gluten free
15	آسیایی	ارزان	شرق شهر	Vegeterian
16	ایرانی	ارزان	شمال شهر	Gluten free

پ) اگر بدانیم مقدار واقعی رضایت به ترتیب به صورت $[0, 1, 0, 1, 0]$ باشد، F_1 score را محاسبه کنید.

پرسش ۵

یک مدل دسته‌بند مولد برای K کلاس را در نظر بگیرید به طوری که توزیع پیشین روی کلاس‌ها به صورت $p(C_k) = \pi_k$ باشد و احتمال متعلق بودن به دسته‌ی k برحسب ویژگی‌ها به صورت $P(\phi | C_k)$ نشان دهیم که ϕ ویژگی‌های داده است. مجموعه‌ی داده‌هایی که در اختیار داریم به صورت زوج‌های $\{\phi_n, t_n\}$ هستند که t_n بردار باینری K بعدی است که اگر متعلق به دسته‌ی j باشد، مولفه‌ی j آن 1 و باقی مولفه‌های آن 0 است. همچنین فرض کنید داده‌ها به صورت مستقل تولید شده‌اند.

الف) نشان دهید جواب بیشینه درست نمایی برای توزیع پیشین به صورت زیر خواهد بود.

$$\pi_k = \frac{N_k}{N}$$

که N تعداد کل داده‌ها و N_k تعداد داده‌های متعلق به دسته‌ی k است. حال فرض کنید توزیع ویژگی داده نسبت به کلاس به صورت زیر باشد

$$P(\phi | C_k) = \mathcal{N}(\phi | \mu_k, \Sigma)$$

ب) تخمین بیشینه درست نمایی برای پارامترهای μ_k و Σ را به دست آورید.

پرسش ۶

سوال عملی

در این سوال می‌خواهیم الگوریتم Adaboost را که شبه‌کد آن در صفحه‌ی قبل نیز آمده، روی یک مجموعه‌ی داده‌ای که به صورت خطی جداپذیر نیست بررسی کنیم. این مجموعه داده، در فایل data.csv قرار گرفته است. الف) اگر دقت کنید برای هر داده، ۲ ویژگی در نظر گرفته شده و هر داده متعلق به یکی از دو کلاس است. ابتدا نمودار این داده‌ها را رسم کنید به گونه‌ای که داده‌های متعلق به یک کلاس رنگ یکسانی داشته باشند. همچنین تقسیم بندی داده‌های تست و آموزش را نیز در این قسمت انجام دهید.

ب) نکته‌ی مهم دیگر این الگوریتم، انتخاب مجموعه‌ی توابع فرض یعنی H است. برای راحتی می‌توانید از کتابخانه‌ی sklearn برای این بخش استفاده کنید. به طور مثال از کلاس DecisionTreeClassifier آن و یا هر کلاس دیگری جز کلاس مربوط به Boosting انتخاب کنید. پس از انتخاب H مراحل یادگیری را مطابق الگوریتم طی کرده تا به تابع نهایی برسید.

پ) دقت (accuracy) تابع به دست آمده را روی مجموعه داده‌ی تست محاسبه کنید. داده‌ها از اینجا قابل دریافت است.