به نام خدا

نظریه یادگیری ماشین

دکتر سیدصالحی

جلسه شانزدهم

دانشکده ریاضی و علوم کامپیوتر

بهار ۱۴۰۳

۱ مقدمه

یک درخت تصمیم‌گیری[۱] یک ساختار داده‌ی سلسله مراتبی است که از استراتژی تجزیه و غلبه[۲] استفاده می‌کند. درخت‌های تصمیم‌گیری از روش‌های غیرپارامتری[۳] به شمار می‌آیند و هم برای مسائل دسته‌بندی[۴] که هم برای مسائل رگرسیون[۵] قابل استفاده می‌باشند. یکی از ویژگی‌های جالب درخت‌های تصمیم قابلیت تفسیرپذیری آن‌هاست. در واقع هر درخت تصمیم را می‌توانیم به سادگی به تعدادی قوانین ساده و قابل فهم تبدیل کنیم. از طرفی درخت‌های تصمیم برای یادگیری زمان زیادی صرف نمی‌کنند و می‌توان از آن‌ها برای مجموعه داده‌های بزرگ استفاده کرد. همچنین از آنجایی که درخت‌های تصمیم قابلیت پشتیبانی از داده‌های عددی[۶] و دسته‌ای[۷] را دارند، برای آموزش آن‌ها به پیش‌پردازش پیچیده‌ای نیاز نداریم.

از آنجایی که درختان تصمیم ارتباط نزدیکی با نظریه اطلاعات[۸] دارند ابتدا نگاهی به مفاهیم کلیدی در این زمینه می‌پردازیم.

۲ نظریه اطلاعات

در سال ۱۹۴۸ شنون[۹] در مقاله‌ای با عنوان A Mathematical Theory of Communication برای اولین به صورت رسمی نظریه اطلاعات را معرفی کرد. در این بخش ما قصد داریم با مفاهیم کلیدی این زمینه

[1] decision tree
[2] divide-and-conquer strategy
[3] nonparametric
[4] classification
[5] regression
[6] numerical data
[7] categorical data
[8] Information Theory
[9] Claude Shannon

مانند اطلاعات[10]، انتروپی اطلاعات[11] و انواع آن، کسب اطلاعات[12] و اطلاعات مشترک[13] آشنا شویم.

۲-۱ اطلاعات

شنون برای تعریف اطلاعات اصول زیر را تعریف کرد:

۱. رویدادی با احتمال وقوع یک کاملا قابل پیش‌بینی بوده و هیچ‌گونه اطلاعی را به همراه ندارد.

۲. هرچه احتمال وقوع یک رویداد کمتر باشد،غیر قابل پیش‌بینی‌تر بوده و اطلاعات بیشتری را به همراه دارد.

۳. اگر دو رویداد مستقل به صورت جداگانه اندازه‌گیری شوند، آنگاه مجموع اطلاعات بدست آمده برابر است با جمع اطلاعات هر یک از رویدادها.

بنابراین اگر اطلاعات حاصل از یک متغیر تصادفی مانند X را با $I(X)$ نشان دهیم، خواهیم داشت:

$$p(x) = 1 \rightarrow I(X) = 0$$
$$p(x) \leq p(y) \rightarrow I(X) \geq I(Y)$$
$$p(x, y) = p(x)p(y) \rightarrow I(X, Y) = I(X) + I(Y)$$

تنها یک خانواده از توابع شرط‌های برگرفته از اصول فوق را محقق می‌کنند و به همین منظور داریم:

$$I(X) := -\log p(x) = \log \frac{1}{p(x)}$$

۲-۲ اِنتروپی

انتروپی یک متغیر تصادفی عبارت از است میزان اطلاعاتی که به صورت متوسط در اختیار ما قرار می‌دهد. یعنی برای یک متغیر تصادفی گسسته نظیر X با تابع جرم احتمال $p : \mathcal{X} \rightarrow [0, 1]$ داریم:

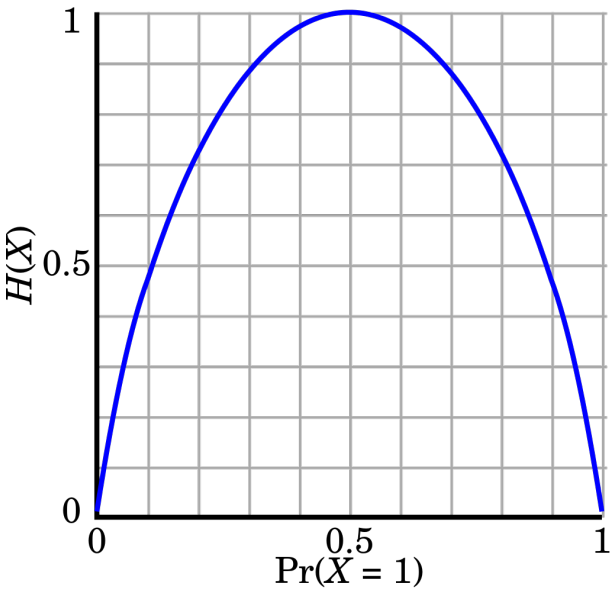
$$H(X) := \mathbb{E}\left[I(X)\right] = -\sum_{x \in \mathcal{X}} p(x) \log p(x)$$

[10] self-information

[11] entropy

[12] information gain, Kullback–Leibler (KL) divergence, relative entropy

[13] mutual information



شکل ۱: هنگامی $p = 1$ است هیچ اطلاعات غیر قابل پیش‌بینی بدست نمی‌آید و مقدار انتروپی برابر با صفر است. به بیانی دیگر با احتمال یک می‌دانیم که $X = 1$ خواهد بود و هیچ عدم قطعیتی نداریم. از طرفی هنگامی که $p = 0$ باشد باز هیچ اطلاعات ارزشمند و غیر منتظره‌ای بدست نیامده چرا که این بار با احتمال یک می‌دانیم که $X = 0$ است و در نتیجه انتروپی دوباره برابر با صفر خواهد شد. بیشترین عدم قطعیت یا انتروپی را هنگامی خواهیم داشت که $p = 0.5$ باشد. در این حالت نمی‌توانیم هیچ تفاوتی میان رویدادهای مختلف قائل شویم و بیشترین عدم قطعیت و انتروپی هنگامی رخ می‌دهد که توزیع احتمالی یکنواخت باشد.

فرض کنید که $X \sim \text{Ber}(p)$ از یک توزیع برنولی پیروی کند. در این صورت نمودار تغییرات انتروپی بر حسب مقدار p را در شکل ۱ مشاهده می‌کنید.

۳-۲ انتروپی توام

انتروپی توام[۱۴] دو متغیر تصادفی نظیر X و Y برابر است با:

$$H(X,Y) = \mathbb{E}\left[-\log p(x,y)\right] = -\sum_{x\in\mathcal{X},\ y\in\mathcal{Y}} p(x,y)\log p(x,y)$$

اگر X و Y از هم مستقل باشند می‌توان این رابطه را به صورت زیر نوشت:

$$H(X,Y) = -\sum_{x\in\mathcal{X},\ y\in\mathcal{Y}} p(x)p(y)\log\left[p(x)p(y)\right] = H(X) + H(Y)$$

[14] joint entropy

۲-۴ انتروپی شرطی

انتروپی شرطی[15] یا عدم قطعیت متغیر تصادفی X به شرط Y برابر است با:

$$H(X|Y) = \mathbb{E}_Y\left[H(X|y)\right] = -\sum_{y\in\mathcal{Y}} p(y) \sum_{x\in\mathcal{X}} p(x|y)\log p(x|y) = -\sum_{x\in\mathcal{X},\ y\in\mathcal{Y}} p(x,y)\log p(x|y)$$

که این رابطه را به صورت زیر نیز می‌توان نوشت:

$$H(X|Y) = -\sum_{x\in\mathcal{X},\ y\in\mathcal{Y}} p(x,y)\log\frac{p(x,y)}{p(y)} = H(X,Y) - H(Y)$$

۲-۵ کسب اطلاعات

میزان کسب اطلاعات از یک متغیر تصادفی نظیر X به شرط مشاهده (یادگیری) ویژگی a برابر است با:

$$IG(X,a) := H(X) - H(X|a)$$

در واقع در اینجا می‌خواهیم بدانیم با دانستن و یا یادگیری یک ویژگی مانند a میزان عدم قطعیت ما نسبت به X به چه میزان کاهش می‌یابد. برای مثال اگر این دو از یکدیگر مستقل باشند در این صورت کسب اطلاعات برابر با صفر می‌شود. این رابطه را می‌توان به صورت واگرایی KL نیز بیان کرد. فرض کنید دو توزیع احتمالاتی گسسته P و Q هر دو بر روی یک فضای نمونه نظیر $\mathcal{X}$ تعریف شده باشند، در این صورت داریم:

$$D_{\mathrm{KL}}(P\|Q) = \sum_{x\in\mathcal{X}} P(x)\log\frac{P(x)}{Q(x)} = -\sum_{x\in\mathcal{X}} P(X)\log Q(x) - H(P)$$

دقت داشته باشید که رابطه فوق نامتقارن بوده و در حالت کلی $D_{\mathrm{KL}}(P\|Q) \neq D_{\mathrm{KL}}(Q\|P)$ است. گاهی عبارت $-\sum_{x\in\mathcal{X}} P(X)\log Q(x)$ را به صورت $H(P,Q)$ نمایش می‌دهند که گرچه مانند انتروپی توام است، اما تعریف آن فرق می‌کند و به آن cross-entropy می‌گویند. بنابراین این رابطه به صورت زیر نیز نمایش داده می‌شود:

$$D_{\mathrm{KL}}(P\|Q) = H(P,Q) - H(P)$$

[15] conditional entropy

که باز هم تاکید می‌شود در این رابطه عبارت $H(P,Q)$ همان cross-entropy بوده و به انتروپی توام ربطی ندارد.

۲-۶ اطلاعات مشترک

اطلاعات مشترک معیاری برای محاسبه میزان وابستگی دو متغیر و به بیانی دیگر نشان دهنده‌ی مقدار اطلاعات بدست آمده در رابطه با یک متغیر تصادفی به شرط مشاهده‌ی دیگری است که به صورت زیر تعریف می‌شود:

$$I(X,Y) := \sum_{x\in\mathcal{X},\ y\in\mathcal{Y}} p(x,y)\log\frac{p(x,y)}{p(x)p(y)} = H(X) - H(Y|X)$$

در نظر داشته باشید که این رابطه متقارن است:

$$I(X,Y) = H(X) - H(X|Y) = H(Y) - H(Y|X) = I(Y,X)$$

۳ درخت تصمیم

یافتن درخت تصمیم بهینه یک مسئله NP-Complete است و برای آموزش آن‌ها از روش‌های حریصانه[۱۶] استفاده می‌کنیم. برای انجام این کار به روش عمل می‌کنیم که در هر مرحله متغیرها را به گونه‌ای که برمی‌گزینیم تا بهترین تقسیم‌بندی را داشته باشیم. فضای فرضیه[۱۷] به گونه‌ای است که می‌تواند ویژگی‌های boolean را به صورت ترکیب فصلی از ترکیب‌های عطفی[۱۸] داده‌ها را به دسته‌های کوچک‌تر تفسیم کند.

۳-۱ الگوریتم $ID3$

الگوریتم ID3[۱۹] در سال ۱۹۸۶ توسط راس کوئینلن[۲۰] معرفی شد. این الگوریتم به روش حریصانه داده‌ها را به صورت بازگشتی به دسته‌های کوچک‌تر تقسیم می‌کند و تا زمانی که تمامی داده‌های درون یک زیردسته دارای برچسب مشترک نباشند و یا ویژگی‌هایی برای تقسیم‌بندی جدید وجود نداشته باشد ادامه می‌دهیم. در هر تقسیم‌بندی به گونه‌ای عمل می‌کنیم که میزان کسب اطلاعات بیشینه باشد و عدم انتروپی یا همان

[16] greedy
[17] hypothesis space
[18] disjunction of conjuctions
[19] Iterative Dichotomiser 3
[20] Ross Quinlan

عدم قطعیت به بیشترین مقدار ممکن کاهش یابد. برای بکارگیری مقدار پیوسته در این الگوریتم باید ابتدا آن‌ها را گسسته کنیم. نحوه عملکرد این الگوریتم به شرح زیر می‌باشد:

الگوریتم ۱ ID3

Require: Examples, Target_Attribute, Attributes
1: Create a root node for the tree
2: **if** all examples are positive **then**
3: **return** the single-node tree Root, with label = +
4: **if** all examples are negative **then**
5: **return** the single-node tree Root, with label = -
6: **if** number of predicting attributes is empty **then**
7: **return** Root, with label = most common value of the target attribute in the examples
8: **else**
9: A = The Attribute that best classifies examples
10: Testing attribute for Root = A
11: **for all** possible values, v_i, of A **do**
12: Add a new tree branch below Root, corresponding to the test $A = v_i$
13: Let Examples(v_i) be the subset of examples that have the value for A
14: **if** Examples(v_i) is empty **then**
15: below this new branch add a leaf node with label = most common target value in the examples
16: **else**
17: below this new branch add subtree **ID3**(Examples(v_i), Target_Attribute, Attributes - $\{A\}$)
18: **return** Root

الگوریتم ID3 از بیش‌برازش رنج می‌برد و با کوچک بودن مجموعه دادگان و یا وجود نویز این پدیده تشدید می‌شود. همچنین ویژگی‌هایی که مقادیر ممکن زیادی دارند نسبت به ویژگی‌های دیگر، حتی اگر حاوی اطلاعات بیشتری باشند، ترجیح داده می‌شوند. الگوریتم‌هایی ارائه شدند که آثار منفی این موارد را تا حدی کاهش دهند. الگوریتم C4.5 تلاش می‌کند تا این مشکلات را تا میزانی برطرف کند. برای حل مسائل رگرسیون به کمک درختان تصمیم الگوریتم CART وجود دارد که به جای استفاده از کسب اطلاعات از معیار Gini impurity استفاده می‌کند.

۳-۲ بیش‌برازش در درختان تصمیم

برای پیش‌گیری از بیش‌برازش در درختان تصمیم روش‌های متعددی وجود دارد که در این بخش به برخی از مهم‌ترین روش‌های موجود می‌پردازیم.

۱. توقف زودهنگام[۲۱]: در این روش هرگاه ادامه الگوریتم کمک شایانی به بهبود الگوریتم نکند، و از نظر

[21] early stopping

آماری تاثیر چندانی نداشته باشد اجرای الگوریتم را متوقف می‌کنیم.

۲. هرس کردن[۲۲]: در این روش ابتدا یک درخت کامل را می‌سازیم و سپس با هرس کردن این درخت تا هنگامی که عملکرد آن بر روی داده‌های صحت‌سنجی[۲۳] منجر به بهبود نشود، اقدام به هرس آن می‌کنیم. در عمل این روش عملکرد بهتری نسبت به توقف زودهنگام دارد.

۳. یادگیری گروهی[۲۴]: روش دیگر برای بهبود عملکرد درخت‌های تصمیم استفاده از یادگیری گروهی است. جنگل‌های تصادفی[۲۵] با استفاده از چند درخت تصمیم به صورت همزمان تلاش می‌کنند تا نتایج بهتری کسب کنند. برای مسائل دسته‌بندی با رای‌گیری و برای مسائل رگرسیون با میانگین‌گیری میان خروجی درخت‌های مختلف سعی می‌کنیم تا به خروجی‌های دقیق‌تری برسیم. در یادگیری ماشین روش‌های یادگیری گروهی بسیار کارمد هستند و از آن‌ها زیاد استفاده می‌شود و شما در جلسات آینده بیشتر با آن‌ها آشنا می‌شود.

[22] pruning
[23] validation
[24] ensemble learning
[25] random forest